

UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR



Grado en Ingeniería Informática

TRABAJO FIN DE GRADO

**Desarrollo de una aplicación web para la gestión de prácticas
externas de la EPS**

Miriam Adrián Ruiz
Tutor: Francisco Jurado Monroy

JULIO 2021

Desarrollo de una aplicación web para la gestión de prácticas externas de la EPS

AUTOR: Miriam Adrián Ruiz

TUTOR: Francisco Jurado Monroy

**Dpto. Ingeniería Informática
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Julio de 2021**

Resumen (castellano)

Hasta la fecha, la gestión de prácticas externas de la Escuela Politécnica Superior de la Universidad Autónoma de Madrid está parcialmente informatizada, por lo que resulta un procedimiento algo rudimentario donde la mayoría de sus gestiones se hacen a través de hilos de correos.

Este Trabajo Fin de Grado ha surgido con la motivación de simplificar y agilizar este procedimiento, cubriendo las necesidades básicas, que se han analizado previamente, facilitando así el trabajo de los coordinadores y empleados de la Oficina de Practicas Externas, y también de los tutores externos que desean que se les apruebe un plan formativo.

Para dar solución a estas carencias se ha desarrollado una aplicación web SPA (*Single Page Application*) con el objetivo de crear una herramienta que sea de fácil acceso para los implicados y mejorar la gestión de solicitudes, pudiendo informatizar el procedimiento en la medida de lo posible. En esta aplicación se podrán crear, visualizar y buscar solicitudes, y además procesar la transición de estados que sufre la solicitud.

Para el desarrollo de este proyecto se ha empleado, en su mayoría, JavaScript, uno de los lenguajes de programación más utilizados hoy en día. Para ello se ha hecho uso de Angular en el lado del cliente y ASP.NET Core en el lado del servidor, donde para la capa de persistencia se ha usado el gestor de base de datos Microsoft SQL Server. Al tratarse de una aplicación SPA, el enrutamiento entre las distintas pantallas se hace gracias al módulo Router de Angular. También se ha hecho uso de componentes del módulo Angular Material para acelerar y simplificar el desarrollo.

A falta de más módulos para desarrollar, cabe señalar que los resultados han sido satisfactorios en cuanto a la creación y uso de la aplicación web.

Abstract (English)

To date, the management of external internships at the Escuela Politécnica Superior of the Universidad Autónoma of Madrid is partially computerised, resulting in a somewhat rudimentary procedure in which most of the procedures are carried out through mail threads.

This Bachelor Thesis has arisen with the aim of simplifying and speeding up this procedure, covering the basic needs, which have been previously analysed, thus facilitating the work of the coordinators and employees of the Internship Office, and also of the external tutors who wish to have a training plan approved.

In order to solve these shortcomings, a web application SPA (Single Page Application) has been developed with the aim of creating a tool that is easy to access for those involved and to improve the management of applications, being able to computerise the procedure as far as possible. In this application it will be possible to create, visualise and search for applications, as well as process the transition of statuses that the application undergoes.

For the development of this project, JavaScript, one of the most widely used programming languages nowadays, has been used for the most part. For this purpose, Angular has been used on the client side and ASP.NET Core on the server side, where the Microsoft SQL Server database manager has been used for the persistence layer. As it is a SPA application, routing between the different screens is done thanks to the Angular Router module. We have also made use of components from the Angular Material module to speed up and simplify development.

In the absence of more modules to develop, it should be noted that the results have been satisfactory in terms of the creation and use of the web application.

Palabras clave (castellano)

Aplicación web, SPA, Angular, ASP.NET, Microsoft SQL Server, componentes, servicios.

Keywords (inglés)

Web application, SPA, Angular, ASP.NET, Microsoft SQL Server, components, services.

Agradecimientos

En primer lugar, agradecer a mi tutor, Francisco, por su motivación constante para la realización de este trabajo y su ayuda hasta el último momento.

A mis padres y mi hermana, que nunca han dejado de creer en mí y gracias a ellos hoy soy quien soy.

A mis compañeros de aventura, Rebeca y Jonathan, que me han hecho mejor persona y un poco menos desastre. Más que compañeros son familia.

Y a mi chico, Miguel, por su apoyo incondicional y siempre estar ahí.

INDICE DE CONTENIDOS

1	Introducción.....	1
1.1	Motivación.....	1
1.2	Objetivos.....	1
1.3	Organización de la memoria.....	1
2	Estado del arte	3
2.1	Procedimiento actual	3
2.2	Aplicaciones web.....	4
2.2.1	Tecnologías en la parte de cliente (frontend)	4
2.2.2	Tecnologías en la parte de servidor (backend)	5
3	Análisis.....	7
3.1	Roles	7
3.2	Casos de uso	7
3.2.1	Caso de uso del administrador.....	7
3.2.2	Caso de uso del coordinador.....	8
3.2.3	Caso de uso del tutor externo	9
3.3	Requisitos	10
3.3.1	Requisitos funcionales.....	10
3.3.2	Requisitos no funcionales.....	11
4	Diseño y desarrollo.....	12
4.1	Estructura.....	12
4.1.1	Capa del navegador	12
4.1.2	Capa del servidor.....	24
4.1.3	Capa de persistencia	24
5	Integración, pruebas y resultados	26
5.1	Caso de uso: Crear solicitud	26
5.2	Caso de uso: Buscar solicitudes	28
5.3	Caso de uso: Enviar solicitud a verificación	30
5.4	Caso de uso: Aprobar solicitud.....	32
5.5	Caso de uso: Asignar alumnos a una solicitud	34
5.6	Caso de uso: Evaluar y cerrar solicitud	36
6	Conclusiones y trabajo futuro.....	38
6.1	Conclusiones.....	38
6.2	Trabajo futuro	39
	Referencias	41
	Glosario	43
	Anexos.....	I
A	Maquetas.....	I

INDICE DE FIGURAS

FIGURA 1. RESULTADOS DEL SONDEO REALIZADO POR STACK OVERFLOW DE TECNOLOGÍAS USADAS EN 2020 [3].....	4
---	---

FIGURA 2. CASO DE USO DEL ROL <i>ADMINISTRADOR</i>	8
FIGURA 3. CASO DE USO DEL ROL <i>COORDINADOR</i>	9
FIGURA 4. CASO DE USO DEL ROL <i>TUTOR EXTERNO</i>	10
FIGURA 5. DIAGRAMA ESTRUCTURA DE LA APLICACIÓN	12
FIGURA 6. DIAGRAMA DE PAQUETES DE LOS COMPONENTES DE LA APLICACIÓN	13
FIGURA 7. COMPONENTE <i>HEADER</i>	14
FIGURA 8. COMPONENTE <i>LOGIN</i>	15
FIGURA 9. COMPONENTE <i>HOME</i>	16
FIGURA 10. COMPONENTE <i>NEW-REQUEST</i> PESTAÑA <i>EMPRESA / PROYECTO FORMATIVO</i>	17
FIGURA 11. COMPONENTE <i>NEW-REQUEST</i> PESTAÑA <i>COMPETENCIAS A OBTENER</i>	17
FIGURA 12. COMPONENTE <i>NEW-REQUEST</i> PESTAÑA <i>CARACTERÍSTICAS DE LA PRÁCTICA</i>	18
FIGURA 13. COMPONENTE <i>REQUEST</i> PESTAÑA <i>EMPRESA / PROYECTO FORMATIVO</i>	19
FIGURA 14. COMPONENTE <i>REQUEST-LIST</i>	20
FIGURA 15. COMPONENTE <i>SEARCH</i>	21
FIGURA 16. COMPONENTE <i>SEARCH</i> MOSTRANDO LOS FILTROS INCLUIDOS EN EL <i>SIDENAV</i>	21
FIGURA 17. CARPETA DONDE SE UBICAN TODOS LOS SERVICIOS.....	22
FIGURA 18. EJEMPLO DE DIÁLOGO EN LA APLICACIÓN	22
FIGURA 19. DIAGRAMA MÁQUINA DE ESTADOS	23
FIGURA 20. DIAGRAMA DE CLASES	25
FIGURA 21. PRUEBA 1 – CASO DE USO <i>CREAR SOLICITUD</i>	27
FIGURA 22. PRUEBA 2 – CASO DE USO <i>CREAR SOLICITUD</i>	27
FIGURA 23. PRUEBA 1 – CASO DE USO <i>BUSCAR SOLICITUDES</i>	28
FIGURA 24. PRUEBA 2 – CASO DE USO <i>BUSCAR SOLICITUDES</i>	29
FIGURA 25. PRUEBA 1 – CASO DE USO <i>ENVIAR A VERIFICACIÓN</i>	30
FIGURA 26. PRUEBA 2 – CASO DE USO <i>ENVIAR A VERIFICACIÓN</i>	31
FIGURA 27. PRUEBA 2 – CASO DE USO <i>ENVIAR A VERIFICACIÓN</i>	31

FIGURA 28. PRUEBA 1 – CASO DE USO <i>APROBAR SOLICITUD</i>	32
FIGURA 29. PRUEBA 2 – CASO DE USO <i>APROBAR SOLICITUD</i>	33
FIGURA 30. PRUEBA 2 – CASO DE USO <i>APROBAR SOLICITUD</i>	33
FIGURA 31. PRUEBA 1 – CASO DE USO <i>ASIGNAR ALUMNOS A UNA SOLICITUD</i>	34
FIGURA 32. PRUEBA 2 – CASO DE USO <i>ASIGNAR ALUMNOS A UNA SOLICITUD</i>	35
FIGURA 33. PRUEBA 2 – CASO DE USO <i>ASIGNAR ALUMNOS A UNA SOLICITUD</i>	35
FIGURA 34. PRUEBA 1 – CASO DE USO <i>EVALUAR Y CERRAR SOLICITUD</i>	36
FIGURA 35. PRUEBA 2 – CASO DE USO <i>EVALUAR Y CERRAR SOLICITUD</i>	37
FIGURA 36. PRUEBA 2 – CASO DE USO <i>EVALUAR Y CERRAR SOLICITUD</i>	37
FIGURA 37. MAQUETA <i>HEADER</i>	I
FIGURA 38. MAQUETA <i>NAV-MENU</i>	I
FIGURA 39. MAQUETA <i>LOGIN</i>	I
FIGURA 40. MAQUETA <i>HOME</i>	II
FIGURA 41. MAQUETA <i>NEW-REQUEST</i> PESTAÑA <i>EMPRESA / PROYECTO FORMATIVO</i>	II
FIGURA 42. MAQUETA <i>NEW-REQUEST</i> PESTAÑA <i>COMPETENCIAS A OBTENER</i>	III
FIGURA 43. MAQUETA <i>NEW-REQUEST</i> PESTAÑA <i>CARACTERÍSTICAS DE LA PRÁCTICA</i>	III
FIGURA 44. MAQUETA <i>REQUEST</i> PESTAÑA <i>EMPRESA / PROYECTO FORMATIVO</i>	IV
FIGURA 45. MAQUETA <i>REQUEST-LIST</i>	IV
FIGURA 46. MAQUETA <i>SEARCH</i>	V
FIGURA 47. MAQUETA <i>SEARCH</i> MOSTRANDO LOS FILTROS INCLUIDOS EN EL <i>SIDENAV</i>	V

INDICE DE TABLAS

TABLA 1. MATRIZ DE TRAZABILIDAD FUNCIONALIDAD/ROL	38
---	----

1 Introducción

1.1 Motivación

Actualmente, el procedimiento para la gestión de prácticas externas (tanto curriculares como extracurriculares) que se emplea en la EPS (Escuela Politécnica Superior) de la UAM (Universidad Autónoma de Madrid) está parcialmente informatizado. La mayoría de los procesos y flujos de información entre los sujetos implicados se realizan a través de hilos de correo. También se requiere de un trabajo adicional para la gestión de todo el control documental y el flujo de trabajo entre los participantes.

La motivación de este Trabajo de Fin de Grado es tratar de simplificar y agilizar el proceso para la gestión de prácticas externas. Para ello se creará una aplicación web que pueda ser utilizada por todos los roles implicados en este procedimiento, pudiendo acceder de manera sencilla a través del navegador.

1.2 Objetivos

Tras la motivación planteada, el objetivo principal de este Trabajo de Fin de Grado es la creación de una aplicación web que facilite el procedimiento de gestión de prácticas externas a los usuarios implicados, que resulte de fácil acceso que permita a los usuarios acceder desde cualquier ordenador y sin necesidad de instalar ningún software en el lado del cliente, más allá de lo necesario en el lado del servidor.

Como objetivo práctico, la aplicación debe ser sencilla e intuitiva, capaz de ejecutarse en cualquier sistema operativo a través de un navegador. En cuanto a la seguridad, la aplicación dispondrá de tres roles: *Administrador*, *Coordinador* y *Tutor externo*, que explicaremos más adelante.

La aplicación será capaz de crear solicitudes, buscarlas y mostrar que tareas pendientes tiene cada usuario según su rol asignado. Los usuarios podrán hacer *login/logout* en esta.

1.3 Organización de la memoria

La memoria consta de los siguientes capítulos:

- **Capítulo 1: Introducción**

En este capítulo se define la motivación y objetivos del proyecto, junto con la organización de la memoria.

- **Capítulo 2: Estado del arte**

En este capítulo se analizarán tanto el procedimiento actual para gestionar las prácticas externas, como las tecnologías empleadas actualmente en la parte de cliente y servidor de las aplicaciones web.

- **Capítulo 3: Análisis**

En este capítulo se analizarán los roles a utilizar, casos de uso por rol y requisitos funcionales y no funcionales.

- **Capítulo 4: Diseño y desarrollo**

En este capítulo se detalla la estructura que mantendrá la aplicación y el desarrollo llevado a cabo para la creación de esta.

- **Capítulo 5: Integración, pruebas y resultados**

En este capítulo se detallan las pruebas realizadas de validación elaboradas sobre los casos de uso más relevantes y sus resultados.

- **Capítulo 6: Conclusiones y trabajo futuro**

En este capítulo se analizan las conclusiones finales a las que se ha llegado con este Trabajo de fin de Grado y los posibles trabajos futuros que se pueden desarrollar para la aplicación.

2 Estado del arte

En este capítulo se describirá el procedimiento actual para la gestión de prácticas externas y las tecnologías más empleadas para el desarrollo de aplicaciones web, haciendo hincapié en las utilizadas para este desarrollo.

2.1 Procedimiento actual

A modo de resumen, el procedimiento actual para la gestión de prácticas que se quiere automatizar es el que se describe a continuación. En primer lugar, una empresa que quiera ofertar prácticas externas se pone en contacto con la Oficina de Prácticas Externas (OPE) de la Escuela Politécnica Superior (EPS). La OPE solicita un plan formativo a la empresa a través de un formulario/documento en Word, en el cual se indican los datos de la empresa, el tutor que tendrá el estudiante en la empresa, las competencias académicas y actividades formativas que se van a realizar dentro de la misma, etc. Una vez que la OPE tiene el plan formativo, se lo reenvía al coordinador de la titulación de las prácticas externas. Este, como persona delegada de por la Comisión de Coordinación de Seguimiento del Título (CCST) debe valorar si el plan formativo tiene coherencia para la titulación que se oferta, si cumple con la cantidad de créditos que el alumno debe superar, si las competencias que se trabajarán están alineadas con las de la titulación, etc. Si la valoración es positiva, el coordinador le asigna una baremación al plan formativo y se lo indica a la OPE. En caso de rechazar la oferta, el coordinador comunica los motivos a la OPE, y esta se lo indicará a la empresa detallando su decisión.

A partir de aquí el plan formativo ya estará dentro del listado de ofertas de prácticas externas. A los alumnos que se matriculen en esta asignatura, se les podrá asignar cualquier oferta de este listado.

Una vez que el alumno se asocia con uno de los planes formativos aprobados para realizar las prácticas, la OPE preparará un convenio que deberá ir firmado por el alumno, un tutor en la empresa y el tutor académico, que suele coincidir con el propio coordinador de titulación. A partir de aquí el alumno quedará asignado a esta oferta y podrá comenzar sus prácticas.

Cuando el alumno finalice su período de prácticas, el tutor en la empresa deberá hacer llegar un informe al coordinador, donde dará valoración de lo realizado en ese periodo por el alumno. El alumno también deberá preparar otro informe comentando las actividades que ha realizado en las prácticas. Con esos dos informes y el seguimiento llevado a cabo por el tutor académico, este último emite el suyo, donde a su vez asignará una nota al alumno, la cual será publicada en Sigma.

Tras analizar el procedimiento actual, puede apreciarse cómo la solución de asistir y automatizar el proceso pasa por la construcción de una aplicación de gestión. En particular, tal y como se ha mencionado en el capítulo anterior, el enfoque por el que se ha apostado en este TFG es el de construir una aplicación Web. En la siguiente sección se verán las tecnologías tanto backend (lado del servidor) como frontend (lado del cliente) que se emplean en la actualidad.

2.2 Aplicaciones web

Actualmente la mayoría de los procedimientos manuales se están informatizando para optimizar el tiempo de trabajo de las personas implicadas, evitar pérdidas de información, y eliminar espacio de almacenamiento en papel, entre otras cosas. La mejor herramienta de que dispone el usuario son las aplicaciones web [1], ya que son fáciles de gestionar, totalmente compatibles con los diferentes navegadores y de fácil acceso sin necesidad de instalar ningún tipo de software. Los usuarios podrán acceder al servidor web por medio de internet mediante un navegador que ejecutará la aplicación. También tienen un bajo consumo de recursos, ya que casi en su totalidad, la aplicación se va a alojar en un servidor web.

En cuanto a las tecnologías a utilizar, hay una gran variedad de combinaciones posibles para las distintas partes. Se detallan a continuación.

2.2.1 Tecnologías en la parte de cliente (frontend)

Según el sondeo de 2020 de la página *Stack Overflow* (<https://stackoverflow.com>), una de las páginas más consultadas por los desarrolladores y cuyo resultado queda reflejado en la Figura 1, JavaScript [2] vuelve a ser, por octavo año consecutivo, el lenguaje de programación más utilizado.

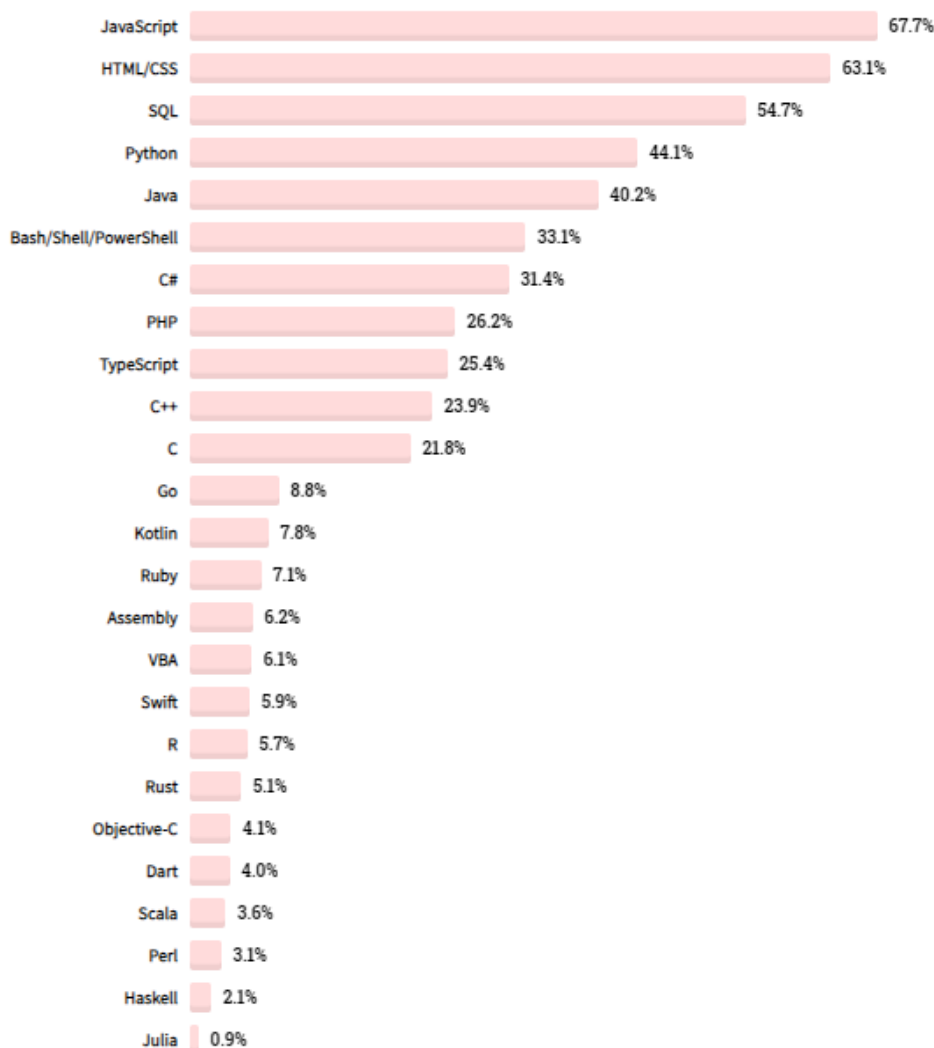


Figura 1. Resultados del sondeo realizado por Stack Overflow de tecnologías usadas en 2020 [3]

El lenguaje JavaScript, junto con el lenguaje de marcas HTML [4] y el de definición de estilos CSS [5], permite dotar de más dinamismo a las aplicaciones web y una mayor interactividad con el usuario. Páginas como Google, Facebook o YouTube, usan este lenguaje como programación frontend. Otra de las características que podemos destacar es que existen una gran cantidad de frameworks [6] que agilizan y facilitan el desarrollo, reutilizando el código ya existente para evitar así la duplicidad del mismo. Con esto el desarrollo será más claro y mantenible por otros desarrolladores. Algunos de los frameworks más populares basados en JavaScript son Angular [7], React [8], Vue [9], Ember [10], Backbone [11], Meteor [12] o Mithril [13].

Como se detallará en subsiguientes capítulos, para el desarrollo del presente TFG se ha utilizado Angular para el desarrollo del cliente web de esta aplicación.

Angular es un framework desarrollado en TypeScript [14], un superconjunto de JavaScript/ECMAScript [15], utilizado para crear y mantener aplicaciones web de tipo SPA. Esto significa que toda la aplicación web se carga una única vez, haciendo que la interacción sea mucho más rápida y evitando el enrutamiento a otras páginas. Su desarrollo se basa en componentes autocontenidos y reutilizables, con el uso de etiquetas HTML personalizadas. No sigue un MVC (*Modelo Vista Controlador*) clásico, ya que el concepto base de Angular es *two-way data binding*, es decir, en la vista se podrá modificar el modelo y en el modelo se podrá modificar la vista.

2.2.2 Tecnologías en la parte de servidor (backend)

En cuanto a la parte de servidor, existen numerosas tecnologías para su desarrollo entre las que destacan: Symfony [16], Java EE [17], Django [18], ASP.NET [19].

Symfony es un framework para desarrollar en lenguaje de programación PHP, siendo este de código abierto. Es una buena opción para sitios web y aplicaciones que necesitan ser escalables. Su sistema se basa en un diseño de componentes modular, lo que le hace flexible y extensible. Además, es compatible con la mayoría de las bases de datos como MySQL, Oracle, PostgreSQL, etc. El mayor inconveniente, aunque existe gran documentación, es la complejidad para familiarizarse con este framework.

Java EE es un framework para desarrollar en lenguaje de programación Java, también de código abierto y multiplataforma. Permite desarrollar aplicaciones portables, robustas y seguras, pero estas características harán complicado su desarrollo. También es necesario que el lado del cliente sea capaz de instalar y ejecutar la máquina virtual Java (*JVM*), para que los programas Java puedan ser ejecutados.

Django es un framework para desarrollar en lenguaje de programación Python, también de código abierto y multiplataforma. El desarrollo de este framework es rápido y limpio, pudiendo crear aplicaciones de gran rendimiento, seguras y escalables. El código de Django es fácil de leer y mantener, pero al ser un framework antiguo, está programado en un único bloque, lo que lo hace más lento en su ejecución y maneja peticiones de una en una.

ASP.NET es un framework de código abierto comercializado por Microsoft. Es ampliamente utilizado por desarrolladores para la creación y el mantenimiento de sitios web dinámicos, aplicaciones web y servicios web XML [20]. Este fue creado para resolver las limitaciones que presentaba ASP (ya que no era un lenguaje orientado a objetos y solo se podía desarrollar con el lenguaje VBScript [21]) mejorando el rendimiento, rapidez en la programación y la seguridad.

Las aplicaciones web creadas con este lenguaje en el lado servidor, se pueden ejecutar en todos los navegadores web y permiten escribir código usando cualquier lenguaje admitido por el ASP.NET Framework, como pueden ser C# [22], VisualBasic.Net [23] o JScript.Net [24].

En el caso del presente TFG, se ha utilizado ASP.NET para el desarrollo del lado del servidor web de esta aplicación, ya que se trata de un framework escalable y con soporte continuo. Además, existe una gran cantidad de documentación al alcance del desarrollador. Ofrece mayor flexibilidad gracias a su soporte MVC. También tiene una comunicación estable con la base de datos SQL Server y gracias a la actualización de ASP a ASP.NET, le hace un framework más rápido y seguro.

Existen dos implementaciones de ASP.NET para compilar aplicaciones del lado de servidor: ASP.NET Framework [25] y ASP.NET Core [26].

ASP.NET Framework no es de código abierto en su totalidad y solo puede ejecutarse en Windows. Está programado en un único bloque, lo que hace difícil su mantenimiento y actualización. Si existiera una nueva versión habría que actualizar todo el bloque. Esta implementación se distribuye por medio de paquetes descargables los cuales se instalan a nivel de equipo de forma única, por lo que sólo puede existir un único paquete a nivel de equipo.

ASP.NET Core comparte la base arquitectónica con ASP.NET Framework, pero muchos de sus módulos han sido reescritos. Es de código abierto y multiplataforma. Se trata de un framework modular, ya que está formado por distintos módulos distribuidos a través de paquetes NuGet [27], haciéndolo más ágil y eficiente. En el caso de haber una nueva versión de algún módulo, únicamente habría que actualizar ese módulo en lugar de todo el bloque, como pasa con ASP.NET Framework. El rendimiento es mayor que el de su predecesor, ya que ha sido siempre una prioridad en su desarrollo. La creación de proyectos con ASP.NET Core se puede realizar en cualquier IDE, cosa que con ASP.NET Framework únicamente podría hacerse con el IDE Visual Studio [28]. También se distribuye por medio de paquetes descargables, pero no solo puede estar instalado a nivel de equipo, si no también a nivel de usuario o de aplicación, siendo así más flexible. Al tener la posibilidad de estar instalado en diferentes niveles, pueden coexistir distintas versiones sin problema.

Por todo lo descrito de estas dos implementaciones de ASP.NET, se ha utilizado ASP.NET Core para el desarrollo del lado del servidor web de esta aplicación.

3 Análisis

En este capítulo se detallará el análisis realizado de los roles, casos de uso y requisitos para el diseño y desarrollo posterior de la aplicación web, la cual gestionará las prácticas externas.

3.1 Roles

En esta aplicación se va a dar soporte a cuatro roles:

- Administrador
- Coordinador
- Tutor externo
- Alumno

Los tres primeros roles van a poder acceder a la aplicación. El último rol no tendrá acceso, pero si será necesario dentro de la aplicación para poder asignar usuarios con el rol alumno a la solicitud.

3.2 Casos de uso

En este apartado se van a definir los casos de uso de los roles administrador, coordinador y tutor externo.

3.2.1 Caso de uso del administrador

El administrador podrá ser capaz de ver sus datos de usuario y las tareas que tiene pendientes en cuanto a las solicitudes.

Por otro lado, se encargará de verificar las solicitudes, asignar alumnos a las solicitudes y reabrir las en caso de haber sido rechazadas. Para realizar todos estos cambios, tendrá que ser capaz también de ver el detalle de las solicitudes.

Por último, también podrá hacer uso del buscador de solicitudes que contiene la aplicación, para consultar cualquier solicitud que desee. Cabe destacar que este rol podrá ver todas las solicitudes creadas en la aplicación.

El diagrama de casos de uso relacionado con lo que acaba de explicarse puede verse detallado en la Figura 2.

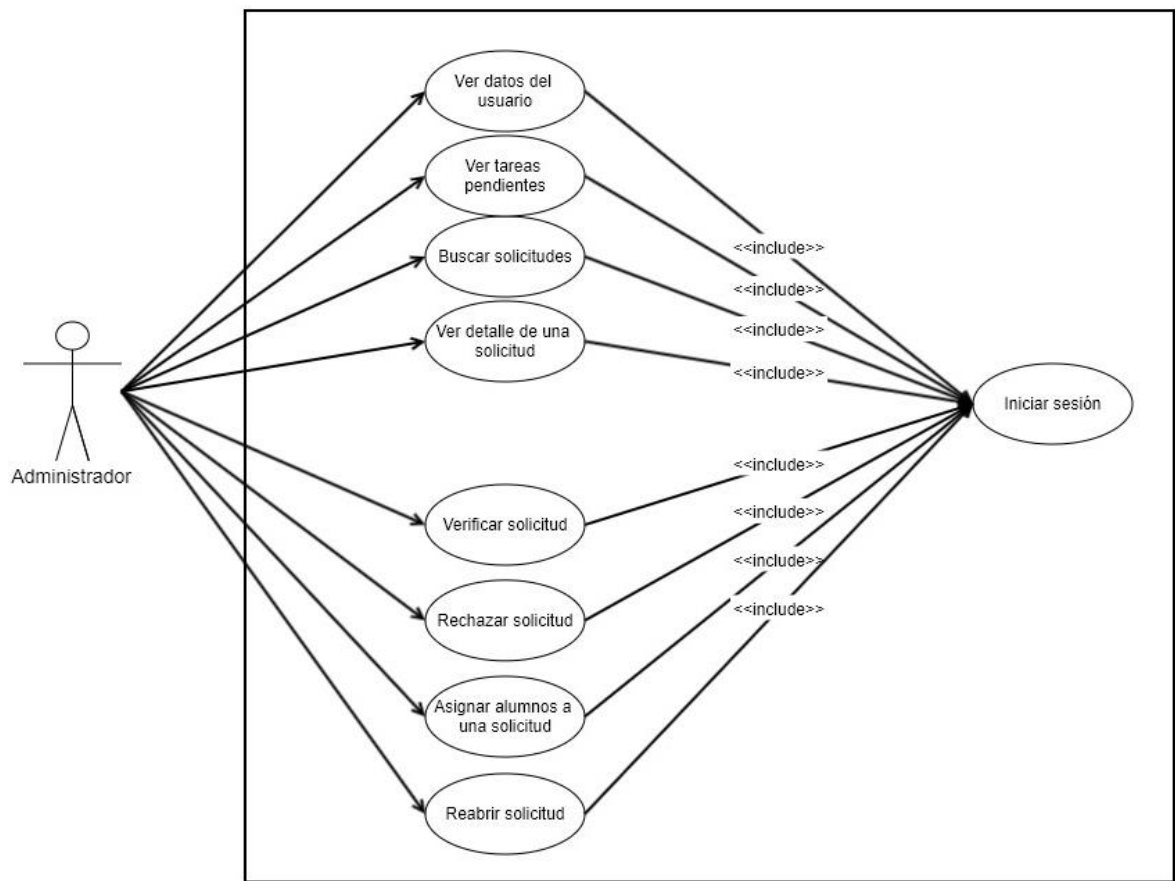


Figura 2. Caso de uso del rol *Administrador*

3.2.2 Caso de uso del coordinador

El coordinador, al igual que el administrador, podrá ser capaz de ver sus datos de usuario y las tareas que tiene pendientes en cuanto a las solicitudes.

También se encargará de aprobar o rechazar las solicitudes, dependiendo de si el plan formativo que se ha facilitado es apto o no. Además, podrá, cuando finalicen las prácticas externas del alumno, evaluarlas y cerrarlas, para así concluir las mismas. Todo esto teniendo en cuenta que podrá ver el detalle de las solicitudes.

El coordinador igualmente podrá hacer uso del buscador de solicitudes que contiene la aplicación para consultar solicitudes. Las solicitudes que estén en estado *Pendiente de Verificar*, no podrá verlas ya que estas únicamente las ve el administrador.

En la Figura 3 puede verse detallado el diagrama de casos de uso de lo descrito anteriormente.

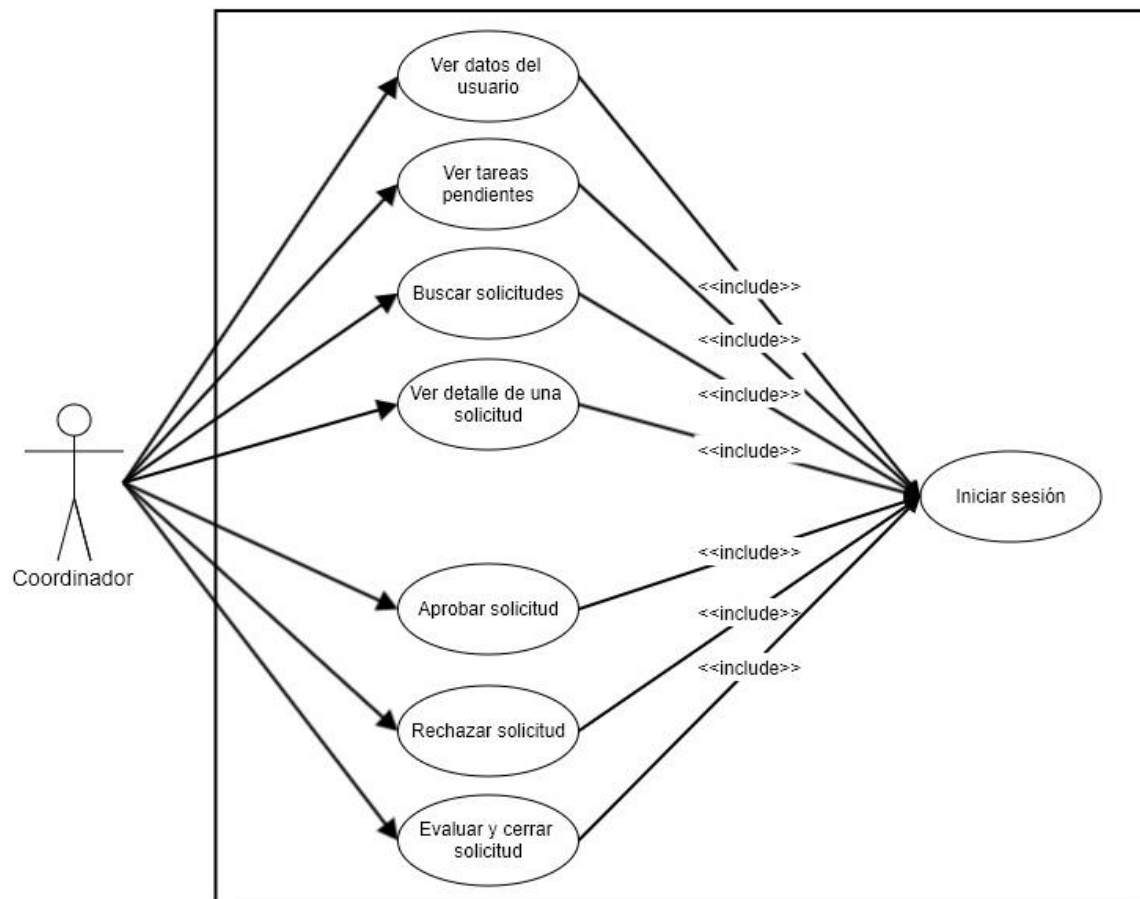


Figura 3. Caso de uso del rol *Coordinador*

3.2.3 Caso de uso del tutor externo

El tutor externo, al igual que los otros dos anteriores, podrá ser capaz de ver sus datos de usuario y las tareas que tiene pendientes en cuanto a las solicitudes.

También se encargará de crear desde la aplicación una nueva solicitud para las prácticas externas.

Además, podrá consultar en el buscador de solicitudes de la aplicación, únicamente las solicitudes creadas por su compañía.

La Figura 4 muestra el detalle del diagrama de casos de uso que acaba de explicarse para este rol.

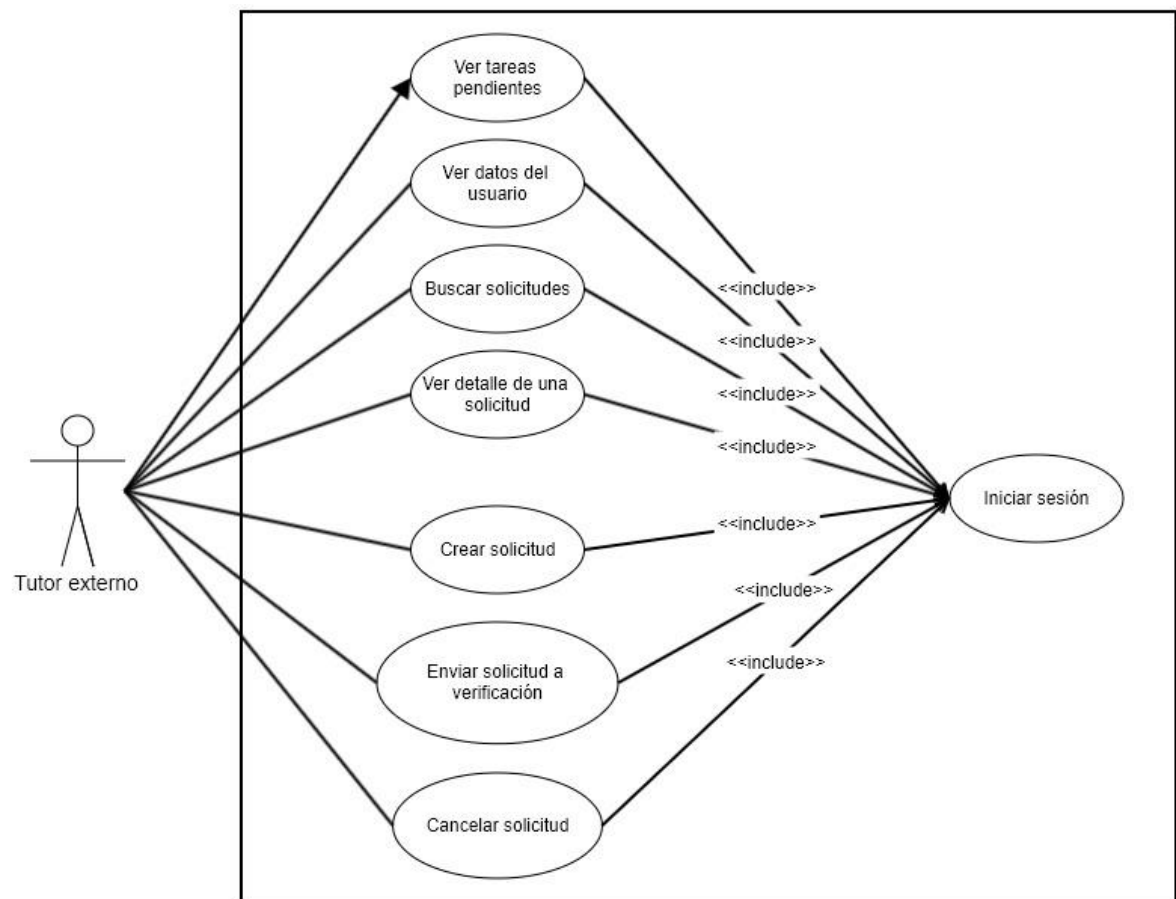


Figura 4. Caso de uso del rol *Tutor externo*

3.3 Requisitos

En este apartado vamos a enumerar todos los requisitos de la aplicación, tanto funcionales como no funcionales.

3.3.1 Requisitos funcionales

Para tratar los requisitos funcionales se va a hacer una distinción entre los tres roles que pueden acceder a la aplicación.

- Cualquier usuario que tenga acceso a la aplicación
 - Login. Acceso a la aplicación a través de un usuario y contraseña proporcionado por el administrador.
 - Logout. Cierre de sesión en la aplicación.
 - Ver sus datos de usuario. Cada usuario que acceda a la aplicación podrá ver su nombre, email y rol que tiene.
 - Ver tareas pendientes. En la pantalla principal de la aplicación cada usuario tendrá un listado de tareas pendientes que tiene que completar.
 - Buscar solicitudes. Todos los usuarios de la aplicación podrán hacer uso del buscador de solicitudes, aunque dependiendo del rol, podrán ver unas solicitudes u otras.
 - Ver detalle de una solicitud. Si el rol puede acceder a esa solicitud podrá ver el detalle de la solicitud.

- **Administrador**
 - Buscar solicitudes. Este rol tendrá acceso a todas las solicitudes.
 - Verificar solicitudes recién llegadas.
 - Rechazar solicitudes recién llegadas.
 - Asignar alumnos a solicitudes aprobadas.
 - Reabrir solicitudes rechazadas.
- **Coordinador**
 - Buscar solicitudes. Este rol podrá ver todas las solicitudes excepto las que estén en estado *Pendiente de verificar*.
 - Aprobar solicitudes ya verificadas.
 - Rechazar solicitudes ya verificadas.
 - Evaluar y cerrar solicitudes ya concluidas.
- **Tutor externo**
 - Crear solicitud.
 - Ver solicitudes de la empresa asociada. En la pantalla principal, a parte de las tareas pendientes, el rol tutor externo podrá ver un listado de todas las solicitudes creadas por su empresa.
 - Buscar solicitudes. Este rol solo podrá ver solicitudes de su propia empresa.

3.3.2 Requisitos no funcionales

- **Portabilidad.** La aplicación se podrá ejecutar en cualquier sistema operativo tanto en Windows, como Linux y Mac y en cualquier navegador como Google Chrome, Firefox, Internet Explorer, etc.
- **Mantenibilidad.** El desarrollo de la aplicación será fácilmente mantenible gracias a su componentización, ya que cada componente es independiente lo que favorece el mantenimiento del mismo. Además, añadir nueva funcionalidad deberá ser sencillo gracias a esta característica.
- **Seguridad.** Solo podrán acceder a la aplicación los usuarios que estén dados de alta en la base de datos y que hayan iniciado sesión previamente en el sistema.
- **Interfaz y usabilidad.** Se creará una aplicación con facilidad de uso, con una interfaz de usuario sencilla e intuitiva.

4 Diseño y desarrollo

En este capítulo se detallará el diseño planteado y su posterior desarrollo para la creación de la aplicación web, en la cual los usuarios accederán a un servidor web mediante un navegador.

4.1 Estructura

Las aplicaciones web normalmente se estructuran como aplicaciones de tres capas: capa del navegador, capa del servidor y capa de persistencia.

Para el desarrollo del presente TFG se ha optado por las siguientes tecnologías:

- En el lado del cliente: Angular.
- En el lado del servidor: ASP.NET Core.
- Para la capa de persistencia: Microsoft SQL Server.



Figura 5. Diagrama estructura de la aplicación

4.1.1 Capa del navegador

En la capa del navegador lo que se pretende es mostrar los datos almacenados en la base de datos. Esta capa se ha estructurado de la siguiente forma:

- Módulos
- Componentes
- Servicios

Un módulo en Angular es una agrupación formada por componentes (bloque de código reutilizable que engloba una funcionalidad específica), directivas (funciones que extienden la funcionalidad del HTML), pipes (funciones que permiten transformar los datos visualmente) y servicios (archivos que proporcionan funcionalidad común entre los distintos componentes o hacen peticiones mediante API REST a la base de datos). También define las dependencias con otros módulos que se necesiten importar y los componentes, directivas o pipes que se puedan exportar.

Los módulos que forman la aplicación son *app.module* y *material.module*. En el módulo *app.module* se incluye todo lo relacionado con la aplicación. También va a incluir las rutas por las que se puede navegar. En el módulo *material.module* se importan todos los módulos de Angular Material [29], para poder así usar la funcionalidad y componentes de este.

Como hemos visto en el capítulo 2, el framework de Angular nos permite desarrollar componentes autocontenidos y reutilizables. Todos los componentes que se han desarrollado para esta aplicación están formados por tres ficheros:

- .TS: Contiene la funcionalidad de la pantalla.
- .HTML: Contiene la estructura de la pantalla.
- .CSS: Contiene los estilos de la pantalla.

En el desarrollo hemos creado los siguientes componentes dependiendo de su funcionalidad:

- Header
- Nav-menu
- Login
- Home
- New-request
- Request
- Request-list
- Search
- Services
- Dialogs
 - Dialog-approve
 - Dialog-assign
 - Dialog-cancel
 - Dialog-evaluate-close
 - Dialog-reject
 - Dialog-reopen
 - Dialog-send-to-approval
 - Dialog-send-to-verified

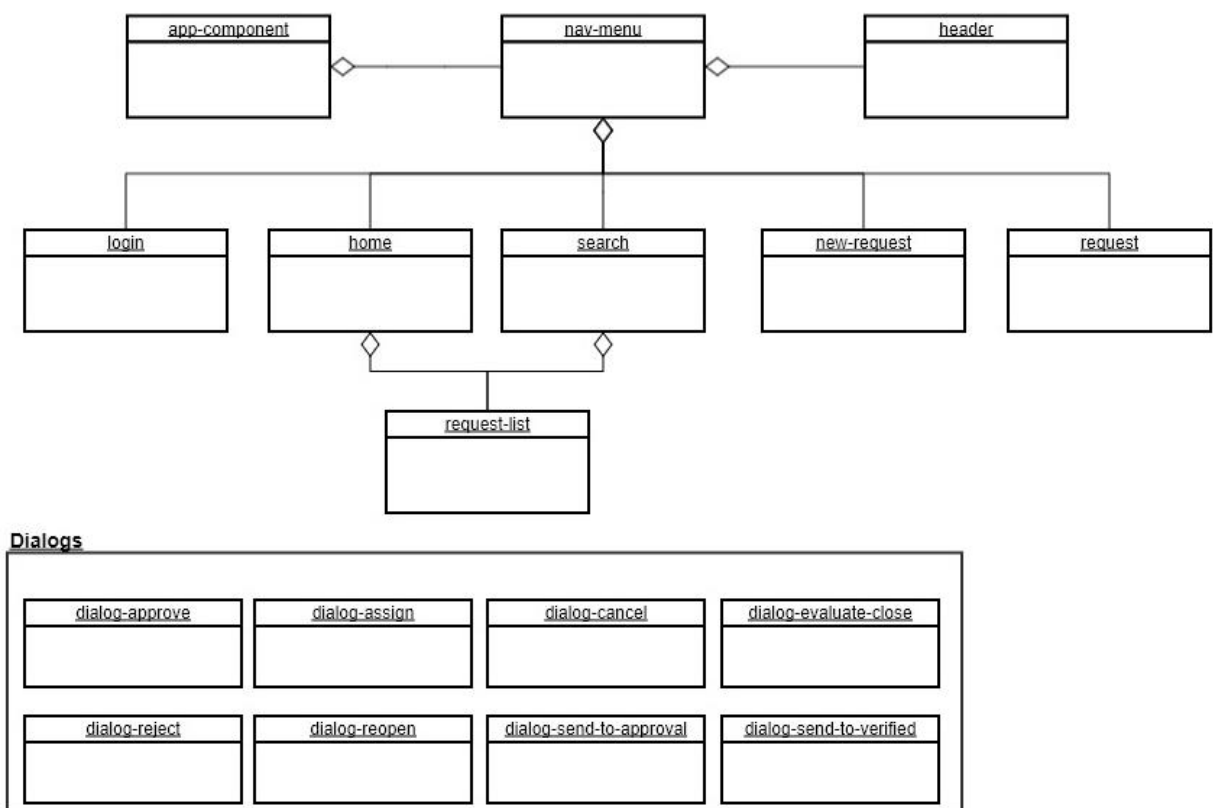


Figura 6. Diagrama de paquetes de los componentes de la aplicación

En los siguientes apartados se explicarán todos los componentes y servicios que forman parte de la aplicación.

Header

Muestra la cabecera de la aplicación web, la cual se visualizará en todo momento una vez hecho *login*, iniciar sesión en la aplicación. La cabecera se compone del logo de la aplicación, la navegación, datos del usuario actual y el logo de la UAM, como se puede ver en la Figura 7.

Este componente está compuesto por los ficheros *header.component.ts*, *header.component.html* y *header.component.css*.

Su funcionalidad principal es contener la navegación de la aplicación, gracias a la cual el usuario va a navegar por las distintas pantallas de esta. Otra funcionalidad es poder hacer *logout*, cerrar sesión en la aplicación. También contiene los logos de la aplicación y de la UAM, y los datos del usuario actualmente conectado.



Figura 7. Componente *Header*

Nav-menu

Va a contener la estructura de la aplicación web, donde se incluyen tanto el componente *header* como el contenido principal, el cual va a ir cambiando según se vaya navegando por resto de componentes de la aplicación.

Este componente está implementado en los ficheros *nav-menu.component.ts*, *nav-menu.component.html* y *nav-menu.component.css*.

Su funcionalidad principal es contener la estructura de la aplicación. Está compuesto por el componente *header* y por una etiqueta especial HTML, *router-outlet*, la cual sirve para mostrar dinámicamente las distintas pantallas, incrustando el componente adecuado para cada ruta por la que vamos navegando. Este componente también contiene el *loading* de la aplicación.

Login

Muestra la pantalla donde el usuario inicia sesión en la aplicación web, tal y como se puede apreciar en la Figura 8.

Este componente está compuesto por los ficheros *login.component.ts*, *login.component.html* y *login.component.css*.

Su funcionalidad principal es autenticar al usuario en la aplicación web.

Cuando se crea un usuario en la base de datos introduce las credenciales para ese nuevo usuario (nombre de usuario y contraseña). Para guardar la contraseña se almacena en la base de datos el cifrado de la misma, segmentado en bloques de bits SALT [30] y HASH [31]. Primero se generan una serie de bits aleatorios, llamados SALT. Después, con el algoritmo HASH, se transforman la contraseña y los bits del SALT en un nuevo bloque de bits arbitrario, el cual va a tener siempre la misma longitud.

Para autenticar al usuario se busca con su nombre de usuario en la base de datos el bloque de bits SALT generado en su creación. Si con el algoritmo HASH se genera el mismo bloque de bits que está guardado en la base de datos la contraseña será correcta.

Una vez validadas las credenciales del usuario, se almacena, de manera local en el navegador, el detalle de este en la propiedad *localStorage* [32]. Esta se ha configurado para que la propiedad expire en un día.



Figura 8. Componente *Login*

Home

Muestra la pantalla principal de la aplicación. Como se puede ver en la Figura 9, en esta pantalla se incluyen los datos del usuario actual, tareas asignadas y solicitudes en el caso del rol *Tutor externo*.

Este componente está compuesto por los ficheros *home.component.ts*, *home.component.html* y *home.component.css*.

Su funcionalidad principal es presentar, de manera rápida y de fácil visualización, tanto los datos del usuario actual, como las tareas pendientes que tiene que realizar. Dependiendo del rol que tenga el usuario, se mostraran las solicitudes en un estado u otro.

- Administrador: solicitudes en estados *Pendiente de verificar*, *Pendiente de asignar* y *Rechazado*.
- Coordinador: solicitudes en estados *Pendiente de aprobar*, *Pendiente de Evaluar*.
- Tutor externo: solicitudes en estados *Borrador* y *Reabierto*.

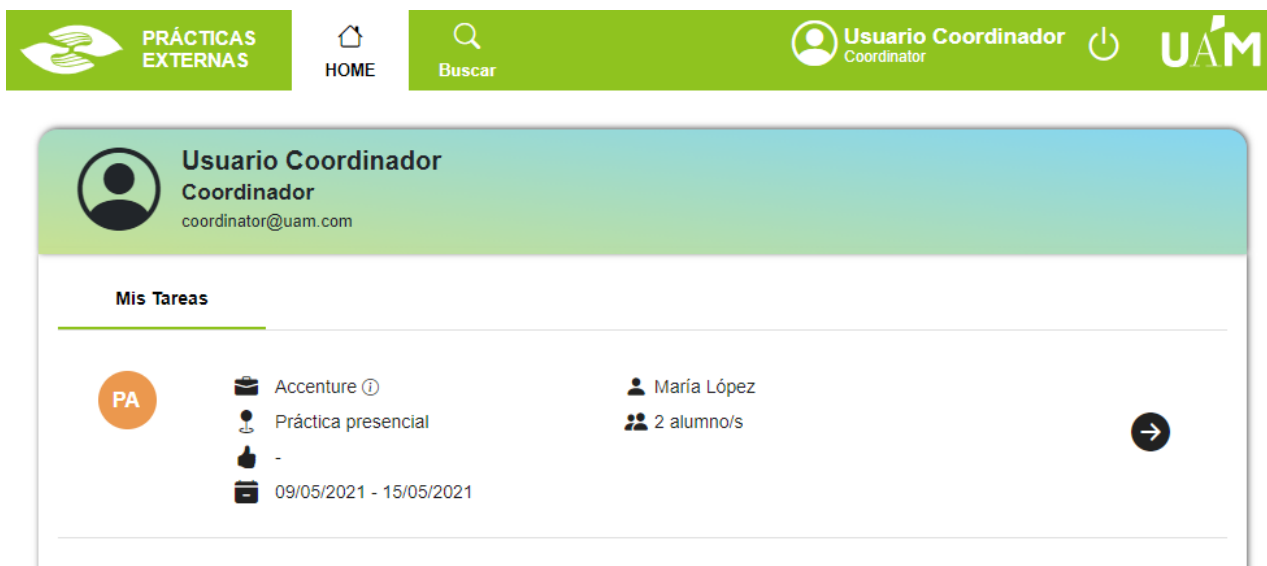


Figura 9. Componente *Home*

New-Request

Como se puede ver en la Figura 10, Figura 11 y Figura 12, muestra la pantalla del formulario para la creación de una nueva solicitud.

Este componente está compuesto por los ficheros *new-request.component.ts*, *new-request.component.html* y *new-request.component.css*.

Su funcionalidad principal es la de crear una nueva solicitud en la base de datos. Para ello, se han separado los datos del formulario en tres pestañas, según las características de cada uno:

- Empresa/Proyecto formativo
- Competencias a obtener
- Características de la práctica

Este componente contiene una botonera donde el *Tutor externo* podrá decidir que hacer con la solicitud:

- Guardar: la solicitud se guardará con estado *Borrador*.
- Enviar a verificación: la solicitud se guardará con estado *Pendiente de verificar*, donde el administrador decidirá si la solicitud sigue adelante o no.

También existe un validador para los campos requeridos. Dependiendo de si se desea *Guardar* la solicitud o se quiere *Enviar a verificación* existirán diferentes campos requeridos.


PRÁCTICAS EXTERNAS


HOME


Nueva solicitud


Buscar


Usuario Tutor Externo Mentor



Nueva solicitud

Guardar

Enviar a verificación

Empresa / Proyecto formativo

Competencias a obtener

Características de la práctica

DATOS DE LA EMPRESA

Empresa

Accenture

Tutor empresarial


PROYECTO FORMATIVO


Departamento en que se realizarán las prácticas


Descripción detallada de todas las actividades a realizar por el/la estudiante


Conocimientos previos por parte del/de la estudiante


Figura 10. Componente *New-request* pestaña *Empresa / Proyecto formativo*



PRÁCTICAS EXTERNAS


HOME


Nueva solicitud


Buscar


Usuario Tutor Externo Mentor



Nueva solicitud

Guardar

Enviar a verificación

Empresa / Proyecto formativo

Competencias a obtener

Características de la práctica

COMPETENCIAS A OBTENER

Selecciona las competencias a obtener por el alumno:

☐ Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación y normativa vigente.

☐ Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social.

☐ Capacidad para comprender la importancia de la negociación, los hábitos de trabajo efectivos, el liderazgo y las habilidades de comunicación en todos los entornos de desarrollo de software.

☐ Capacidad para elaborar el pliego de condiciones técnicas de una instalación informática que cumpla los estándares y normativas vigentes.

Figura 11. Componente *New-request* pestaña *Competencias a obtener*

The screenshot shows the 'Nueva solicitud' (New request) form in the 'Características de la práctica' (Practice characteristics) tab. The form is divided into two columns. The left column contains fields for 'Lugar de realización de la práctica' (Practice location), 'Fecha inicio' (Start date), 'Fecha fin' (End date), 'Duración' (Duration), and 'Número de estudiantes que solicita' (Number of students requesting). The right column contains fields for 'Tipo de práctica' (Type of practice), 'En caso de semipresencialidad, detallar la planificación' (In case of semi-presentiality, detail the planning), 'Importe de la ayuda al estudio (se recomienda un mínimo de 435€ brutos al mes por 4h/día)' (Study aid amount (it is recommended a minimum of 435€ gross per month for 4h/day)), and 'Nombre de la persona que firmará el anexo' (Name of the person who will sign the annex). The form has a green header with navigation links and a green footer with the UAM logo.

PRÁCTICAS EXTERNAS **HOME** **Nueva solicitud** **Buscar** **Usuario Tutor Externo Mentor** **UAM**

Nueva solicitud

[Guardar](#) [Enviar a verificación](#)

Empresa / Proyecto formativo **Competencias a obtener** **Características de la práctica**

CARACTERÍSTICAS DE LA PRÁCTICA

Lugar de realización de la práctica

Tipo de práctica

Fecha inicio **Fecha fin**

Elige una fecha Elige una fecha

Duración

Número de estudiantes que solicita

En caso de semipresencialidad, detallar la planificación

Importe de la ayuda al estudio (se recomienda un mínimo de 435€ brutos al mes por 4h/día)

Nombre de la persona que firmará el anexo

Figura 12. Componente *New-request* pestaña *Características de la práctica*

Request

Muestra en pantalla el detalle del formulario de una solicitud ya creada, sea cual sea su estado. Esto se puede ver en la Figura 13.

Este componente está compuesto por los ficheros *request.component.ts*, *request.component.html* y *request.component.css*.

Su funcionalidad principal es contener toda la información referente a la solicitud. El componente muestra una cabecera con los datos más relevantes de la solicitud junto con el estado en el que se encuentra.

También, como en el componente *new-request*, contiene una botonera donde los botones aparecerán dependiendo del estado actual de la solicitud y del rol con el que se acceda en ese momento a la aplicación.

PRÁCTICAS EXTERNAS HOME Nueva solicitud Buscar Usuario Tutor Externo Mentor UAM

Detalle de la solicitud

Accenture ⓘ Victor Rodriguez
Práctica presencial 1 alumno/s ⓘ
A - Excelente
01/06/2021 - 30/09/2021

CERRADO

Empresa / Proyecto formativo Competencias a obtener Características de la práctica

DATOS DE LA EMPRESA

Empresa ⓘ Tutor empresarial
Accenture Victor Rodriguez

PROYECTO FORMATIVO

Departamento en que se realizarán las prácticas
Departamento Sharepoint

Descripción detallada de todas las actividades a realizar por el/la estudiante
Aprendizaje de SharePoint. Programación de aplicaciones Angular y SharePoint.

Figura 13. Componente *Request* pestaña *Empresa / Proyecto formativo*

Request-list

Muestra una vista general de un listado de solicitudes. Esto se puede ver en la Figura 14.

Este componente está compuesto por los ficheros *request-list.component.ts*, *request-list.component.html* y *request-list.component.css*.

Su funcionalidad principal es listar las solicitudes, mostrando los datos más relevantes de estas. Este componente se inyecta en otros componentes como *home* o *search*, como se puede ver en la Figura 15.

Cada solicitud del listado contiene un botón que nos llevará al detalle de la solicitud.

El listado de solicitudes que se muestran en este componente se ha pasado a través de otro componente gracias a la propiedad *@Input* [33], la cual se usa para recibir datos en un componente. El componente origen pasa el dato a través de una propiedad de entrada DOM, incluida en la plantilla (HTML), por medio de binding usando corchetes. El componente destino recibe el dato en el campo donde tiene la propiedad *@Input*.

























	 Accenture ⓘ  Práctica presencial  -  18/03/2021 - 18/07/2021	 María López  2 alumno/s	
	 Accenture ⓘ  Práctica telemática  -  20/02/2021 - 07/06/2021	 Victor Rodríguez  2 alumno/s	
	 Accenture ⓘ  Práctica presencial y telemática  -  18/05/2021 - 20/08/2021	 María López  1 alumno/s	

Figura 14. Componente *Request-list*

Search

Muestra la pantalla de búsqueda de solicitudes en la aplicación, la cual tendrá un *sidenav* con una serie de filtros para aplicar en la búsqueda. Esta funcionalidad se puede ver en la Figura 15 y Figura 16.

Este componente está compuesto por los ficheros *search.component.ts*, *search.component.html* y *search.component.css*.

Su funcionalidad principal es buscar solicitudes en la base de datos, aplicando los filtros que se deseen. Dependiendo del rol que tenga el usuario que acceda a la aplicación se mostrarán unas u otras solicitudes.

Los usuarios con rol de *Tutor externo* de una empresa no podrán ver las solicitudes creadas por otra empresa diferente. Los usuarios con rol *Coordinador* tampoco podrán ver las solicitudes con estado *Pendiente de verificar*. Los usuarios con rol *Administrador* podrán ver todas las solicitudes creadas en la base de datos.

PRÁCTICAS EXTERNAS

HOME

Nueva solicitud

Buscar

Usuario Tutor Externo Mentor

UAM

19 solicitudes encontradas

C	Accenture ⓘ Práctica presencial - 18/03/2021 - 18/07/2021	María López 2 alumno/s	→
PA	Accenture ⓘ Práctica telemática - 20/02/2021 - 07/06/2021	Víctor Rodríguez 2 alumno/s	→
A	Accenture ⓘ Práctica presencial y telemática - 18/05/2021 - 20/08/2021	María López 1 alumno/s	→

Figura 15. Componente *Search*

PRÁCTICAS EXTERNAS

HOME

Nueva solicitud

Filtros búsqueda

19 solicitudes encontradas

C	Accenture ⓘ Práctica presencial - 18/03/2021 - 18/07/2021
PA	Accenture ⓘ Práctica telemática - 20/02/2021 - 07/06/2021
A	Accenture ⓘ Práctica presencial y telemática - 18/05/2021 - 20/08/2021
C	Accenture ⓘ Práctica presencial y telemática

FECHA INICIO
Borrar Filtro

FECHA FIN
Borrar Filtro

Elige una fecha

Elige una fecha

ESTADO SOLICITUD
Borrar Filtro

VALORACIÓN SOLICITUD
Borrar Filtro

TIPO DE SOLICITUD
Borrar Filtro

ALUMNO
Borrar Filtro

TUTOR EXTERNO
Borrar Filtro

BUSCAR

Figura 16. Componente *Search* mostrando los filtros incluidos en el *sidemav*

Services

Agrupación de todos los servicios del proyecto, tanto los que proporcionan funcionalidad común entre los distintos componentes como los que hacen peticiones mediante API REST a la base de datos. Esto se puede ver en la Figura 17.

En cuanto a estos servicios habría que destacar *execute-action.service.ts*. Este servicio contiene la funcionalidad relacionada con la transición de estados de las solicitudes. Otro servicio importante es *authentication.service.ts*, el cual contiene la funcionalidad referente al usuario actual, con funciones importantes como *login* y *logout*. El servicio *reference-data.service.ts* contiene una función que devuelve un objeto con todos los datos maestros de la base de datos. El servicio *request.service.ts* contiene todo lo relacionado con la creación, guardado y validación de las solicitudes.

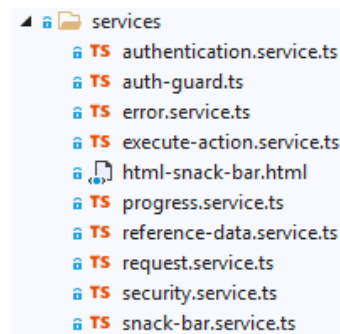


Figura 17. Carpeta donde se ubican todos los servicios

Dialogs

Agrupación de todas las pantallas que muestran un diálogo, los cuales se encargan de los cambios de estado de la solicitud.

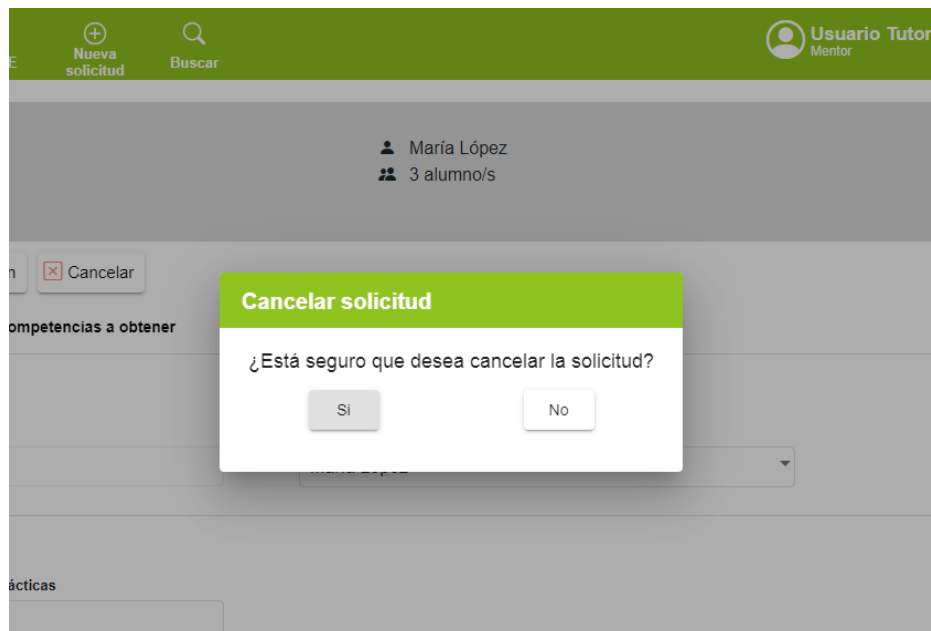


Figura 18. Ejemplo de diálogo en la aplicación

La funcionalidad principal de estos diálogos es reflejar la transición de estados de las solicitudes que puede verse en la Figura 19.

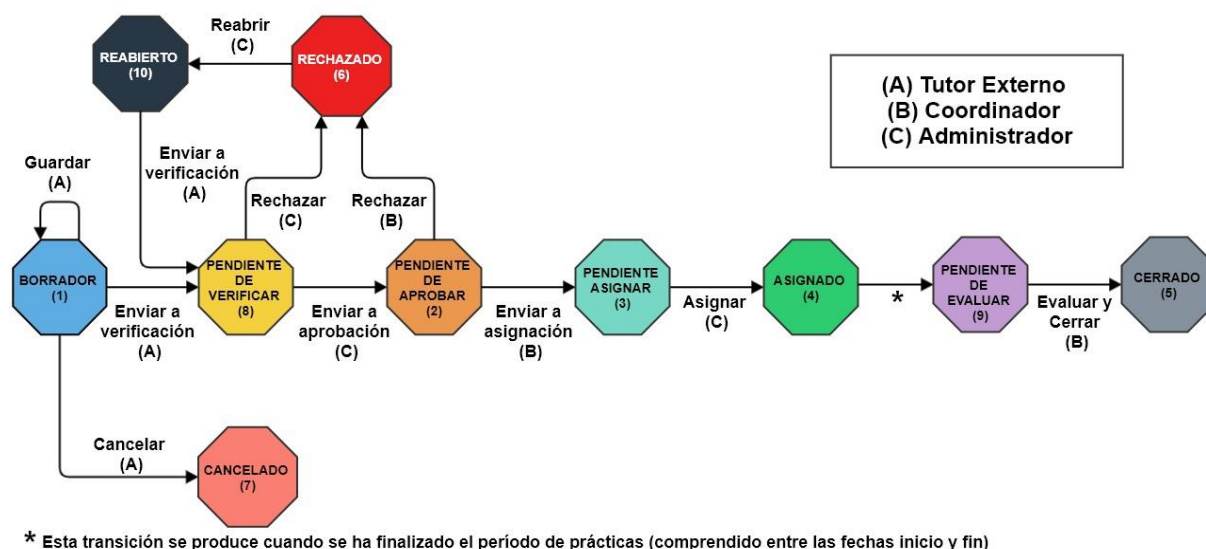


Figura 19. Diagrama máquina de estados

Estos se abren al pinchar en los botones del detalle de la solicitud. Ahora detallaremos cada uno de los diálogos:

- Dialog-approve: Diálogo que se muestra al pulsar el botón *Aprobar*. Dentro del diálogo, al pulsar en *Cancelar* se cierra el diálogo, y en *Aceptar* la solicitud pasa al estado *Aprobado*.
- Dialog-assign: Diálogo que se muestra al pulsar el botón *Asignar*. Dentro del diálogo, al pulsar en *Cancelar* se cierra el diálogo, y en *Aceptar* la solicitud pasa al estado *Asignado*. Es obligatorio asignar tantos usuarios como número de alumnos se requieran en la solicitud.
- Dialog-cancel: Diálogo que se muestra al pulsar el botón *Cancelar*. Dentro del diálogo, al pulsar en *Cancelar* se cierra el diálogo, y en *Aceptar* la solicitud pasa al estado *Cancelado*. Esto puede hacerlo el usuario con rol *Tutor Externo* y únicamente cuando una solicitud está en estado *Borrador* o *Reabierto*.
- Dialog-evaluate-close: Diálogo que se muestra al pulsar el botón *Evaluar y Cerrar*. Dentro del diálogo, al pulsar en *Cancelar* se cierra el diálogo, y en *Aceptar* la solicitud pasa al estado *Cerrado*. Es obligatorio evaluar a cada alumno asignado en la solicitud.
- Dialog-reject: Diálogo que se muestra al pulsar el botón *Rechazar*. Al pulsar a *Cancelar* se cierra el diálogo, y en *Aceptar* la solicitud pasa al estado *Rechazado*. Es obligatorio poner rellenar la razón por la que se ha rechazado esa solicitud, para así el *Tutor externo* pueda comprender el porqué de esa decisión.
- Dialog-reopen: Diálogo que se muestra al pulsar el botón *Reabrir*. Al pulsar a *Cancelar* se cierra el diálogo, y en *Aceptar* la solicitud pasa al estado *Reabierto*.
- Dialog-send-to-approval: Diálogo que se muestra al pulsar el botón *Enviar a aprobación*. Al pulsar a *Cancelar* se cierra el diálogo, y en *Aceptar* la solicitud pasa al estado *Pendiente de aprobar*.
- Dialog-send-to-verified: Diálogo que se muestra al pulsar el botón *Enviar a verificación*. Al pulsar a *Cancelar* se cierra el diálogo, y en *Aceptar* la solicitud pasa al estado *Pendiente de verificar*.

4.1.2 Capa del servidor

En la capa del servidor lo que se pretende es acceder a los datos almacenados en la base de datos para después poderlos mostrar en la capa del navegador. En el desarrollo hemos distribuido esta capa de la siguiente manera:

- Entities
- Controllers
- Services

Entities

Namespace que contiene todas las clases C# referentes a las tablas de la base de datos y las clases complementarias necesarias para el modelado de la aplicación.

Controllers

Namespace que contiene todas las clases C# que realizan peticiones API REST en el servidor.

Services

Namespace que contiene todas las clases C# con funcionalidad común que permiten el acceso a la base de datos. Existe una clase especial, *DataService*, que contiene todos los repositorios de conexión a la base de datos.

La comunicación entre el servidor y la base de datos se hace a través del *EntityLite* [34], ORM de código abierto y ligero, para usar en aplicaciones escritas en C# que acceden a bases de datos relacionales.

4.1.3 Capa de persistencia

En la capa de persistencia lo que se pretende es almacenar los datos en una base de datos. Las bases de datos más utilizadas son las relacionales. Existen diversos SGBD (*Sistemas Gestores de Bases de Datos*) [35] donde se nos permite crear y administrar bases de datos. Los SGBD más utilizados son MySQL [36], Oracle [37], PostgreSQL [38] y Microsoft SQL Server [39]. También es popular MongoDB [40], aunque se trate de una base de datos no relacional.

Se ha utilizado Microsoft SQL Server para la implementación de la base de datos.

Microsoft SQL Server es un SGBD relacionales basado en el lenguaje Transact-SQL [41] (una implementación del estándar ANSI del lenguaje SQL). Su propietario es Microsoft. Se caracteriza por su escalabilidad, estabilidad, seguridad y alta disponibilidad. También destaca por ser un potente entorno gráfico de administración. Además, es multiplataforma y permite administrar información de otros servidores de datos. A su vez, soporta transacciones y procedimientos almacenados.

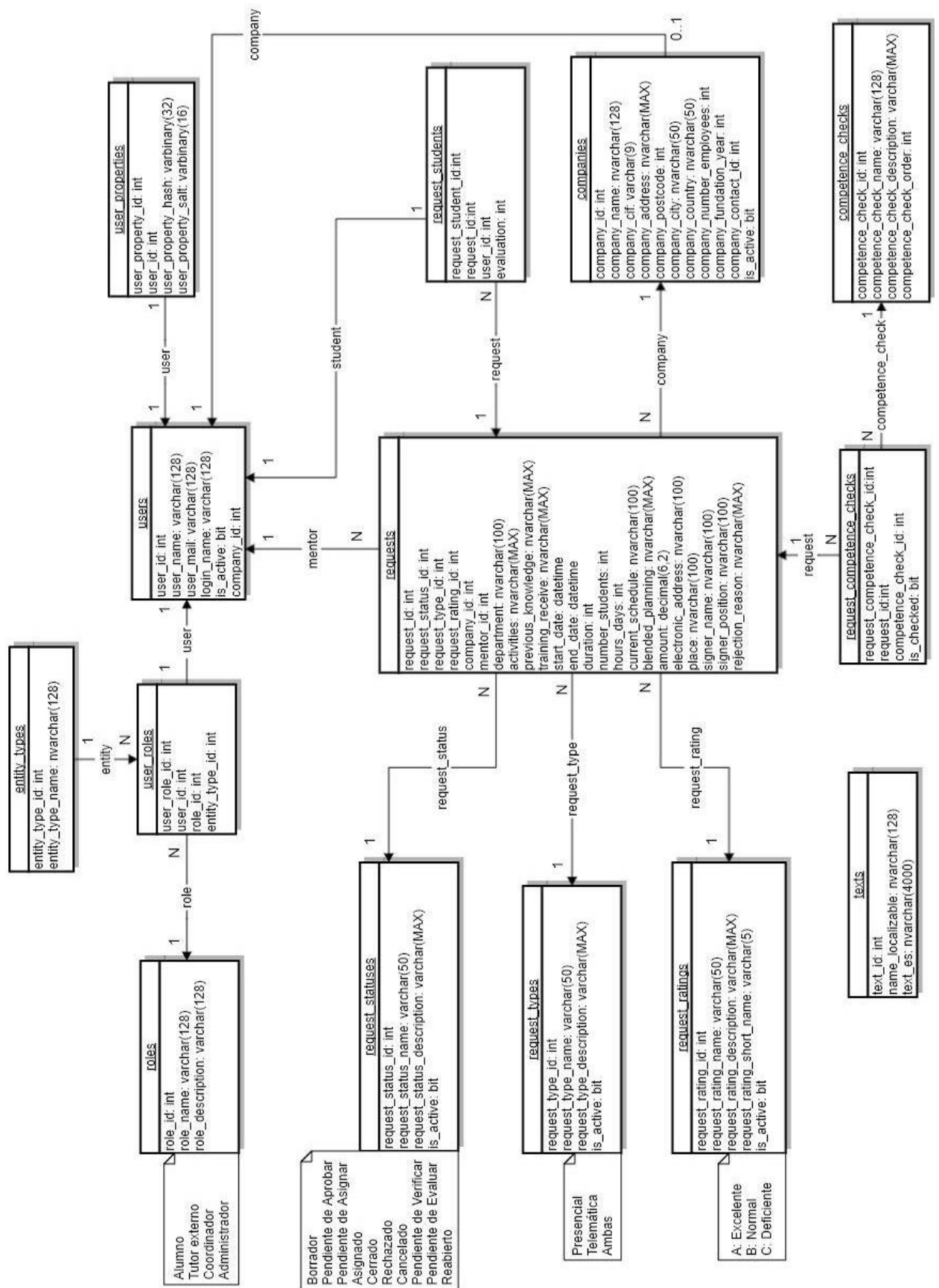


Figura 20. Diagrama de clases

5 Integración, pruebas y resultados

Atendiendo al tipo de aplicación desarrollada para el presente TFG, se ha estimado que las pruebas de validación son el mejor modo de comprobar si se han cubierto los requisitos planteados inicialmente. En particular, en este capítulo se detallarán las pruebas de validación realizadas sobre los casos de uso más relevantes.

5.1 Caso de uso: Crear solicitud

Actor primario: Tutor externo.

Descripción: Este caso de uso permite a los usuarios con rol *Tutor externo* crear nuevas solicitudes en la aplicación con estado *Borrador*.

Interesados:

- Administrador, Coordinador: desean tener nuevas solicitudes para que los alumnos puedan hacer prácticas externas.
- Tutor externo: desea crear una nueva solicitud.

Precondiciones: El usuario debe haber iniciado sesión en la aplicación.

Garantía de éxito (postcondiciones): El usuario con rol *Tutor externo* crea una nueva solicitud y esta se almacena en estado *Borrador*.

Escenario principal de éxito:

1. El usuario accede a la pantalla *Nueva solicitud*.
2. El usuario rellena los campos requeridos para guardar la solicitud.
3. El usuario pincha en el botón *Guardar*.
4. El usuario ha creado una nueva solicitud en la aplicación con estado *Borrador*.
5. La aplicación te lleva a la pantalla *Request* para ver el detalle de la solicitud creada.

Extensiones:

- 2a. El usuario no rellena algún campo requerido. Al pinchar en el botón *Guardar*, el sistema muestra en la pantalla un mensaje de error '*Hay campos incompletos*', listando los campos que faltan por rellenar.

Requerimientos cumplidos: Permitir la creación de una nueva solicitud en la aplicación en estado *Borrador*.

- Prueba 1: No rellenamos ningún campo.

The screenshot shows the 'Nueva solicitud' (New request) form. The top navigation bar includes 'PRÁCTICAS EXTERNAS', 'HOME', 'Nueva solicitud', 'Buscar', 'Usuario Tutor Externo Mentor', and 'UAM'. The form has three tabs: 'Empresa / Proyecto formativo', 'Competencias a obtener', and 'Características de la práctica'. The 'Empresa / Proyecto formativo' tab is active, showing fields for 'Empresa' (filled with 'Accenture') and 'Tutor empresarial'. A green modal box with an information icon and the text 'HAY CAMPOS INCOMPLETOS:' is displayed over the form, listing the following missing fields: 'Tutor empresarial', 'Lugar de realización de la práctica', 'Fecha inicio', 'Fecha fin', 'Número de estudiantes que solicita (Mínimo 1)', and 'Conocimientos previos por parte del/de la estudiante'. A 'Cerrar' button is at the bottom of the modal. Below the modal, the 'PROYECTO FORMATIVO' section is visible with fields for 'Departamento en que se realizarán las prácticas', 'Descripción detallada de todas las actividades del estudiante', and 'Conocimientos previos por parte del/de la estudiante'.

Figura 21. Prueba 1 – Caso de uso *Crear solicitud*

- Prueba 2: Rellenamos únicamente los campos requeridos.

The screenshot shows the 'Detalle de la solicitud' (Request details) page. The top navigation bar is the same as in Figure 21. The page header shows 'Detalle de la solicitud' and a 'BORRADOR' button. The request details are displayed in a grey box: 'Empresa' (Accenture), 'Tutor empresarial' (María López), 'Número de estudiantes' (2 alumno/s), and 'Fecha' (13/06/2021 - 14/10/2021). Below this, there are three buttons: 'Guardar', 'Enviar a verificación', and 'Cancelar'. The 'Empresa / Proyecto formativo' tab is active, showing the 'DATOS DE LA EMPRESA' section with 'Empresa' (Accenture) and 'Tutor empresarial' (María López) fields. The 'PROYECTO FORMATIVO' section is also visible with the 'Departamento en que se realizarán las prácticas' field.

Figura 22. Prueba 2 – Caso de uso *Crear solicitud*

5.2 Caso de uso: *Buscar solicitudes*

Actor primario: Administrador, Coordinador, Tutor externo.

Descripción: Este caso de uso permite a los usuarios buscar solicitudes en la aplicación, usando los filtros que sean necesarios. Dependiendo del rol que acceda les mostrará unas u otras solicitudes.

Interesados:

- Administrador, Coordinador, Tutor externo: desean buscar solicitudes en la aplicación.

Precondiciones: El usuario debe haber iniciado sesión en la aplicación.

Garantía de éxito (postcondiciones): El usuario obtiene una lista con las solicitudes que cumplen los requisitos especificados en su búsqueda.

Escenario principal de éxito:

1. El usuario accede a la pantalla *Buscar*.
2. El usuario introduce los filtros necesarios para buscar la solicitud.
3. El usuario pincha en el botón *Buscar*.
4. El usuario encuentra las solicitudes que cumplen los criterios.

Extensiones:

- 4a. El usuario no encuentra ninguna solicitud que cumpla los criterios. La pantalla *Buscar* no le muestra la solicitud que estaba buscando.

Requerimientos cumplidos: Permitir la búsqueda de una solicitud por el usuario.

Cuestiones pendientes: Eliminar filtros en la pantalla de búsqueda. Pagar el listado.

- Prueba 1: Rellenamos filtros sin encontrar la solicitud.

The screenshot shows the 'Buscar' (Search) interface. At the top, there's a navigation bar with 'PRÁCTICAS EXTERNAS', 'HOME', and 'Nueva solicitud'. Below this, the main area is divided into two sections. The left section, which is currently empty, shows '0 solicitudes encontradas' and 'No hay tareas para mostrar'. The right section is a 'Filtros búsqueda' (Search Filters) panel. It contains several filter categories: 'FECHA INICIO' and 'FECHA FIN' (both with 'Elige una fecha' and a calendar icon), 'ESTADO SOLICITUD' (currently set to 'Rechazado'), 'VALORACIÓN SOLICITUD', 'TIPO DE SOLICITUD', 'ALUMNO', and 'TUTOR EXTERNO'. Each category has a 'Borrar Filtro' (Clear Filter) button. At the bottom right of the filter panel is a green 'BUSCAR' button.

Figura 23. Prueba 1 – Caso de uso *Buscar solicitudes*

- Prueba 2: Rellenamos filtros encontrando la solicitud deseada.

PRÁCTICAS EXTERNAS **HOME** **Nueva solicitud** **Buscar**

1 solicitudes encontradas

A **Accenture** **Práctica presencial y telemática** **18/05/2021 - 20/08/2021**

Filtros búsqueda

FECHA INICIO **Borrar Filtro** **FECHA FIN** **Borrar Filtro**

Elige una fecha **Elige una fecha**

ESTADO SOLICITUD **Borrar Filtro**

Asignado

VALORACIÓN SOLICITUD **Borrar Filtro**

TIPO DE SOLICITUD **Borrar Filtro**

ALUMNO **Borrar Filtro** **TUTOR EXTERNO** **Borrar Filtro**

BUSCAR

Figura 24. Prueba 2 – Caso de uso *Buscar solicitudes*

5.3 Caso de uso: *Enviar solicitud a verificación*

Actor primario: Tutor externo.

Descripción: Este caso de uso permite a los usuarios con el rol *Tutor externo* crear solicitudes y enviarlas a verificación.

Interesados:

- Administrador: desea verificar solicitudes en la aplicación.
- Tutor externo: desea que su solicitud sea verificada por el administrador.

Precondiciones: El usuario debe haber iniciado sesión en la aplicación.

Garantía de éxito (postcondiciones): El usuario envía a verificar la solicitud que ha creado.

Escenario principal de éxito:

1. El usuario accede a la pantalla *Nueva solicitud*.
2. El usuario rellena todos los campos de la solicitud.
3. El usuario pincha en el botón *Enviar a verificación*.
4. El usuario ha creado una nueva solicitud en la aplicación con estado *Pendiente de Verificar*.
5. La aplicación abre la pantalla *Request* para ver el detalle de la solicitud creada.

Extensiones:

- 2a. El usuario no rellena algún campo requerido. Al pinchar en el botón *Enviar a verificación*, el sistema muestra en la pantalla un mensaje de error ‘*Hay campos incompletos*’, listando los campos que faltan por rellenar.

Requerimientos cumplidos: Permitir la creación de una nueva solicitud en la aplicación en estado *Pendiente de Verificación*.

- Prueba 1: No rellenamos ningún campo.

The screenshot shows the 'Nueva solicitud' (New request) form in a web application. The form is partially filled with 'Accenture' in the 'Empresa' field. A modal dialog box is overlaid on the form, displaying the message 'HAY CAMPOS INCOMPLETOS:' (There are incomplete fields:) followed by a list of required fields: Tutor empresarial, Departamento, Actividades a realizar, Conocimientos previos, Formación a recibir, Competencias a obtener, Lugar de realización de la práctica, Fecha inicio, Fecha fin, Duración, Número de estudiantes que solicita (Mínimo 1), Horas/Día, Horario concreto, Tipo de práctica, Importe de ayuda al estudio, Nombre persona que firma, Cargo persona que firma, and Dirección electrónica. A 'Cerrar' (Close) button is at the bottom of the modal. The background form has sections for 'DATOS DE LA EMPRESA', 'PROYECTO FORMATIVO', and 'CONOCIMIENTOS PREVIOS'.

Figura 25. Prueba 1 – Caso de uso *Enviar a verificación*

- Prueba 2: Rellenamos todos los campos.

The screenshot shows the 'Nueva solicitud' (New request) form. The form is divided into three tabs: 'Empresa / Proyecto formativo', 'Competencias a obtener', and 'Características de la práctica'. The 'Empresa / Proyecto formativo' tab is active, showing fields for 'DATOS DE LA EMPRESA' (Company), 'PROYECTO FORMATIVO' (Training Project), and 'Competencias a obtener' (Competencies to be obtained). The 'Empresa' field contains 'Accenture'. The 'PROYECTO FORMATIVO' field contains 'Departamento SharePoint'. The 'Competencias a obtener' field contains 'Aprendizaje de SharePoint, programación de aplicaciones web Angular + SharePoint'. A green modal box is displayed over the form, asking '¿Está seguro que desea enviar a verificar la solicitud?' (Are you sure you want to send the request for verification?). The modal has two buttons: 'Si' (Yes) and 'No' (No).

Figura 26. Prueba 2 – Caso de uso *Enviar a verificación*

The screenshot shows the 'Detalle de la solicitud' (Request Details) page. The page is divided into three tabs: 'Empresa / Proyecto formativo', 'Competencias a obtener', and 'Características de la práctica'. The 'Empresa / Proyecto formativo' tab is active, showing fields for 'DATOS DE LA EMPRESA' (Company), 'PROYECTO FORMATIVO' (Training Project), and 'Competencias a obtener' (Competencies to be obtained). The 'Empresa' field contains 'Accenture'. The 'PROYECTO FORMATIVO' field contains 'Departamento SharePoint'. The 'Competencias a obtener' field contains 'Aprendizaje de SharePoint, programación de aplicaciones web Angular + SharePoint'. The 'Detalle de la solicitud' section shows the request details: 'Accenture' (Company), 'Práctica presencial' (On-site practice), '15/06/2021 - 16/10/2021' (Dates), 'María López' (Tutor), and '2 alumno/s' (Students). A yellow button labeled 'PENDIENTE DE VERIFICAR' (Pending verification) is visible in the top right corner.

Figura 27. Prueba 2 – Caso de uso *Enviar a verificación*

5.4 Caso de uso: Aprobar solicitud

Actor primario: Coordinador.

Descripción: Este caso de uso permite a los usuarios con el rol *Coordinador* aprobar una solicitud que está en estado *Pendiente de Aprobar* y cambiarla de estado a *Pendiente de Asignar*.

Interesados:

- Coordinador: desea aprobar la solicitud en estado *Pendiente de Aprobar*.
- Tutor externo: desea que aprueben su solicitud para asignar alumnos a las Prácticas Externas.

Precondiciones: El usuario debe haber iniciado sesión en la aplicación.

Garantía de éxito (postcondiciones): El usuario aprueba la solicitud.

Escenario principal de éxito:

1. El usuario accede a la pantalla *Home*.
2. El usuario navega al detalle de la solicitud en estado *Pendiente de Aprobar*.
3. El usuario pincha en el botón *Aprobar*.
4. El usuario indica la valoración de la solicitud y pincha en el botón *Guardar*.
5. El usuario ha aprobado la solicitud y su estado cambia a *Pendiente de Asignar*.

Extensiones:

- 4a. El usuario no rellena la valoración requerida. Al pinchar en el botón *Guardar* el sistema muestra en la pantalla un mensaje de error '*Hay campos incompletos*', listando los campos que faltan por rellenar.

Requerimientos cumplidos: Permitir la aprobación de la solicitud por el usuario indicando la valoración de la misma.

- Prueba 1: No rellenamos el campo de la valoración.

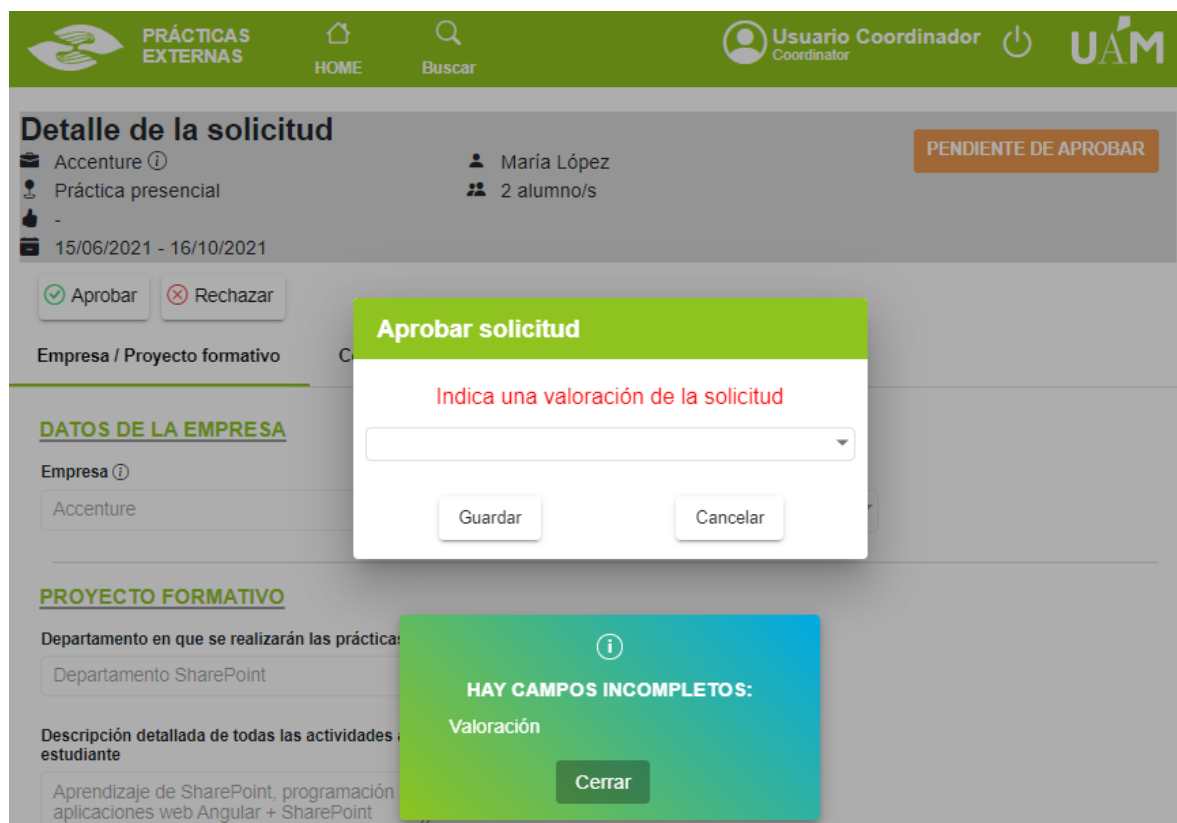


Figura 28. Prueba 1 – Caso de uso *Aprobar solicitud*

- Prueba 2: Rellenamos el campo de la valoración.

The screenshot shows the 'Detalle de la solicitud' page for a request from Accenture. The status is 'PENDIENTE DE APROBAR'. A modal titled 'Aprobar solicitud' is open, prompting the user to 'Indica una valoración de la solicitud'. The dropdown menu is set to 'A - Excelente'. The background page shows details for the company (Accenture), practice type (Práctica presencial), and dates (15/06/2021 - 16/10/2021). There are buttons for 'Aprobar' and 'Rechazar'.

Figura 29. Prueba 2 – Caso de uso *Aprobar solicitud*

The screenshot shows the 'Detalle de la solicitud' page with the status 'PENDIENTE DE ASIGNAR'. The 'DATOS DE LA EMPRESA' section includes the company name 'Accenture' and the 'Tutor empresarial' dropdown set to 'María López'. The 'PROYECTO FORMATIVO' section includes the department 'Departamento SharePoint' and a description of activities: 'Aprendizaje de SharePoint, programación de aplicaciones web Angular + SharePoint'. The page has tabs for 'Empresa / Proyecto formativo', 'Competencias a obtener', and 'Características de la práctica'.

Figura 30. Prueba 2 – Caso de uso *Aprobar solicitud*

5.5 Caso de uso: Asignar alumnos a una solicitud

Actor primario: Administrador.

Descripción: Este caso de uso permite a los usuarios con el rol *Administrador* asignar alumnos a una solicitud que está en estado *Pendiente de Asignar* y cambiarla de estado a *Asignado*.

Interesados:

- Administrador: desea asignar alumnos a la solicitud en estado *Pendiente de Asignar*.
- Tutor externo: desea tener alumnos para que comiencen las Prácticas externas.

Precondiciones: El usuario debe haber iniciado sesión en la aplicación.

Garantía de éxito (postcondiciones): El usuario asigna los alumnos a la solicitud.

Escenario principal de éxito:

1. El usuario accede a la pantalla *Home*.
2. El usuario navega al detalle de la solicitud en estado *Pendiente de Asignar*.
3. El usuario pincha en el botón *Asignar alumnos*.
4. El usuario indica los alumnos que desea asignar y pincha en el botón *Guardar*.
5. El usuario ha asignado los alumnos a la solicitud y su estado cambia a *Asignado*.

Extensiones:

- 4a. El usuario no rellena los alumnos requeridos. Al pinchar en el botón *Guardar* el sistema muestra en la pantalla un mensaje de error '*Hay campos incompletos*', listando los campos que faltan por rellenar.

Requerimientos cumplidos: Permitir la asignación de alumnos a una solicitud por el usuario.

- Prueba 1: No rellenamos ningún campo.

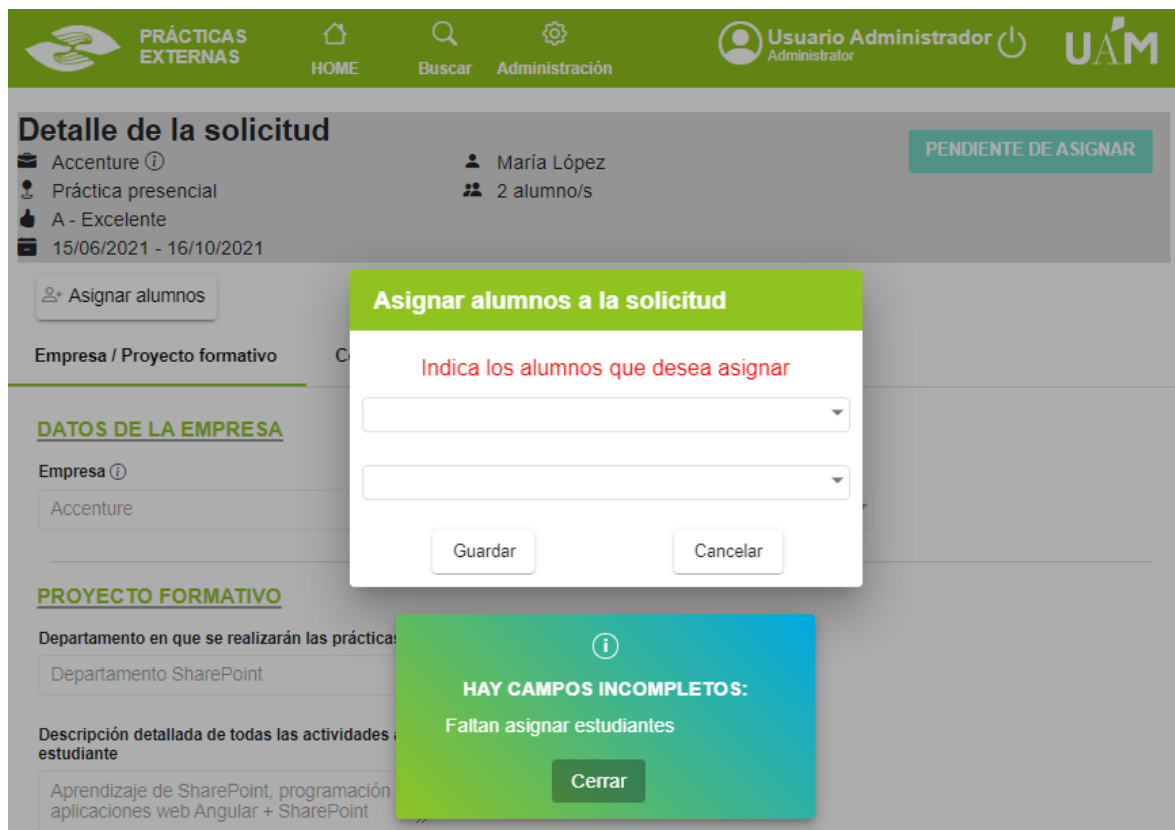


Figura 31. Prueba 1 – Caso de uso *Asignar alumnos a una solicitud*

- Prueba 2: Rellenamos todos los campos.

The screenshot shows the 'Detalle de la solicitud' page for a request from Accenture. The status is 'PENDIENTE DE ASIGNAR'. A modal titled 'Asignar alumnos a la solicitud' is open, prompting the user to 'Indica los alumnos que desea asignar'. Two dropdown menus are visible, with 'Adrián Montero' and 'Alvaro Morena' selected. The background form includes fields for 'Empresa / Proyecto formativo', 'DATOS DE LA EMPRESA' (Empresa: Accenture), 'PROYECTO FORMATIVO' (Departamento: Departamento SharePoint), and a description of activities.

Figura 32. Prueba 2 – Caso de uso *Asignar alumnos a una solicitud*

The screenshot shows the 'Detalle de la solicitud' page after the students have been assigned. The status is now 'ASIGNADO'. A modal titled 'Alumnos asignados' is open, listing 'Adrián Montero' and 'Alvaro Morena' with their email addresses. The background form is the same as in Figure 32, but it also includes a 'Tutor empresarial' field with 'María López' selected.

Figura 33. Prueba 2 – Caso de uso *Asignar alumnos a una solicitud*

5.6 Caso de uso: *Evaluar y cerrar solicitud*

Actor primario: Coordinador.

Descripción: Este caso de uso permite a los usuarios con el rol *Coordinador* evaluar la solicitud y cerrarla.

Interesados:

- Coordinador: desea evaluar la solicitud y cerrarla para su resolución.

Precondiciones: El usuario debe haber iniciado sesión en la aplicación.

Garantía de éxito (postcondiciones): El usuario evalúa la solicitud y la cierra.

Escenario principal de éxito:

1. El usuario accede a la pantalla *Home*.
2. El usuario navega al detalle de la solicitud en estado *Pendiente de Evaluar*.
3. El usuario pincha en el botón *Evaluar y Cerrar*.
4. El usuario indica la nota de cada alumno asignado a la solicitud y pincha en el botón *Guardar*.
5. El usuario ha evaluado la solicitud y su estado cambia a *Asignado*.

Extensiones:

- 4a. El usuario no rellena las notas de los alumnos. Al pinchar en el botón *Guardar* el sistema muestra en la pantalla un mensaje de error '*Hay campos incompletos*', listando los campos que faltan por rellenar.

Requerimientos cumplidos: Permitir la evaluación de alumnos de una solicitud por el usuario.

- Prueba 1: No rellenamos ningún campo.

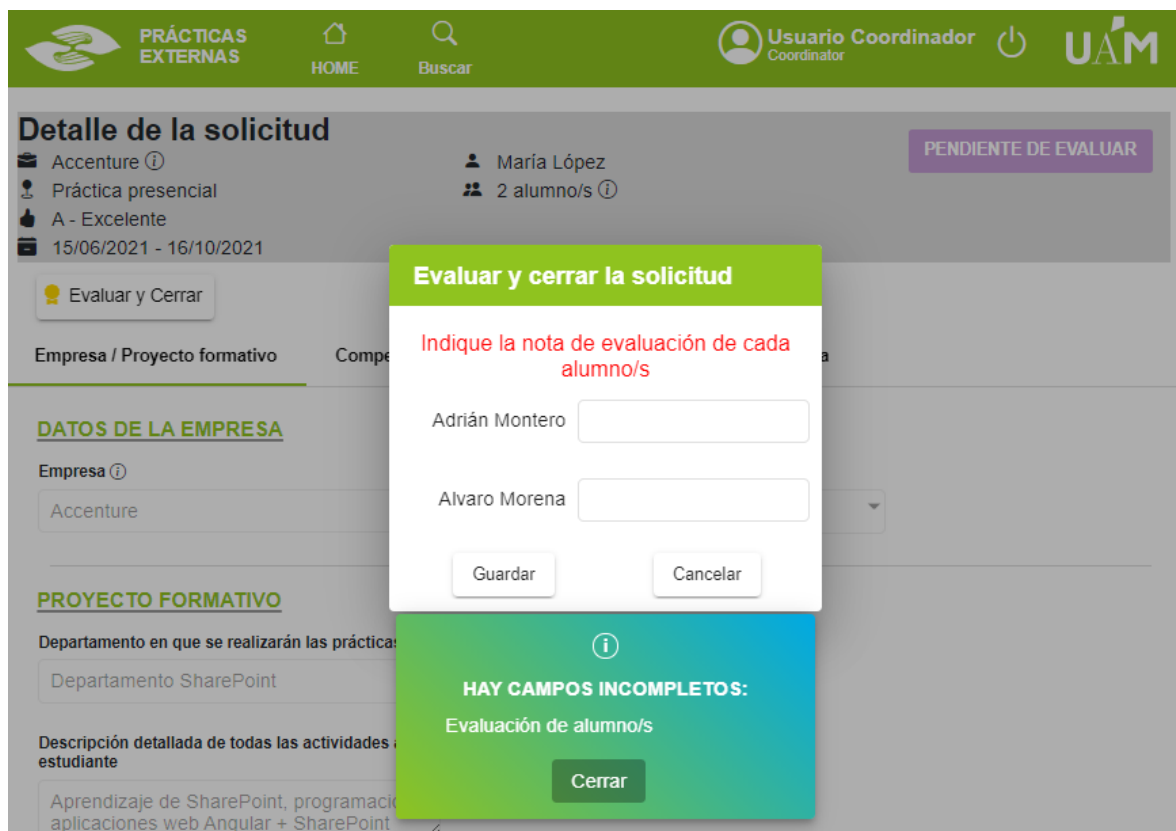


Figura 34. Prueba 1 – Caso de uso *Evaluar y cerrar solicitud*

- Prueba 2: Rellenamos todos los campos.

Detalle de la solicitud

Accenture ⓘ
 Práctica presencial
 A - Excelente
 15/06/2021 - 16/10/2021

María López
 2 alumno/s ⓘ

PENDIENTE DE EVALUAR

Evaluar y cerrar la solicitud

Indique la nota de evaluación de cada alumno/s

Adrián Montero

Alvaro Morena

Guardar Cancelar

DATOS DE LA EMPRESA

Empresa ⓘ
 Accenture

PROYECTO FORMATIVO

Departamento en que se realizarán las prácticas
 Departamento SharePoint

Descripción detallada de todas las actividades a realizar por el/la estudiante
 Aprendizaje de SharePoint, programación de aplicaciones web Angular + SharePoint

Figura 35. Prueba 2 – Caso de uso *Evaluar y cerrar solicitud*

Detalle de la solicitud

Accenture ⓘ
 Práctica presencial
 A - Excelente
 15/06/2021 - 16/10/2021

María López
 2 alumno/s ⓘ

CERRADO

Alumnos asignados

Adrián Montero (adrian.montero@estudiante.uam.es) 8

Alvaro Morena (alvaro.morena@estudiante.uam.es) 7

DATOS DE LA EMPRESA

Empresa ⓘ
 Accenture

Tutor empresarial
 María López

PROYECTO FORMATIVO

Departamento en que se realizarán las prácticas
 Departamento SharePoint

Descripción detallada de todas las actividades a realizar por el/la estudiante
 Aprendizaje de SharePoint, programación de aplicaciones web Angular + SharePoint

Figura 36. Prueba 2 – Caso de uso *Evaluar y cerrar solicitud*

6 Conclusiones y trabajo futuro

En este capítulo se detallarán las conclusiones finales sacadas al realizar este trabajo y las posibles ampliaciones que se podrían desarrollar en un futuro.

6.1 Conclusiones

En el desarrollo de este proyecto he comprobado que las aplicaciones web permiten a los usuarios un acceso rápido a los datos sin necesidad de instalar ningún software más allá del navegador web. Desde el punto de vista del desarrollador, la herramienta ha sido programada teniendo en cuenta principios de escalabilidad para permitir la futura extensión de la funcionalidad.

Las aplicaciones web son versátiles, ya que tienen multitud de posibilidades en la elección de las tecnologías a utilizar tanto en el lado del cliente como en el del servidor. Dependiendo de las características de la aplicación que se desee desarrollar se utilizarán unas tecnologías del lado del cliente junto con otras del lado del servidor. Si se hace una buena elección de estas, se podrá conseguir una aplicación más potente y rápida.

En cuanto a la aplicación creada, en la Tabla 1 se pueden apreciar los roles que se han creado y su acceso a la funcionalidad a través de las distintas pantallas de la aplicación.

Funcionalidad/Rol	Administrador	Coordinador	Tutor externo
Login	X	X	X
Ver datos del usuario	X	X	X
Ver tareas pendientes	X	X	X
Buscar solicitudes	X	X	X
Ver detalle de una solicitud	X	X	X
Crear solicitud			X
Enviar solicitud a verificación			X
Verificar solicitud	X		
Aprobar solicitud		X	
Asignar alumnos a una solicitud	X		
Evaluar y cerrar solicitud		X	
Cancelar solicitud			X
Rechazar solicitud	X	X	
Reabrir solicitud	X		

Tabla 1. Matriz de trazabilidad funcionalidad/rol

Para concluir este Trabajo de Fin de Grado, he comprobado la carga de trabajo que requiere el procedimiento para crear una aplicación web desde cero, incluyendo análisis, diseño, desarrollo y pruebas. Cabe destacar los beneficios que puede aportar a la Universidad Autónoma de Madrid esta aplicación, ya que simplificará y agilizará el proceso para la gestión de prácticas externas.

6.2 Trabajo futuro

Este proyecto incluye un desarrollo con la mínima funcionalidad que se necesita para la gestión de prácticas externas. Existen muchas posibilidades de ampliación de la aplicación.

Una de ellas sería la creación de una pantalla de administración, donde se podrán configurar los usuarios y datos maestros. En esta pantalla sólo accederán los usuarios con rol *Administrador* para crear y editar datos de usuarios, y crear y editar datos maestros, como pueden ser las compañías.

Para los usuarios internos que pertenezcan a la Universidad Autónoma de Madrid se podría incluir el sistema de verificación de usuarios de la misma.

Otra mejora podría ser la posibilidad de tener una gestión de documentos, donde se podrán subir todos los informes necesarios que se han detallado en la sección 2.1.

Referencias

- [1] Aplicación web [último acceso: 18/06/2021] Disponible en: https://es.wikipedia.org/wiki/Aplicaci%C3%B3n_web
- [2] JavaScript [último acceso: 18/06/2021] Disponible en: <https://www.javascript.com/>
- [3] Sondeo 2020 Stack Overflow [último acceso: 18/06/2021] Disponible en: <https://insights.stackoverflow.com/survey/2020#technology-programming-scripting-and-markup-languages>
- [4] HTML [último acceso: 18/06/2021] Disponible en: <https://lenguajehtml.com/html/>
- [5] CSS [último acceso: 18/06/2021] Disponible en: <https://lenguajecss.com/>
- [6] Framework [último acceso: 18/06/2021] Disponible en: <https://es.wikipedia.org/wiki/Framework>
- [7] Angular [último acceso: 18/06/2021] Disponible en: <https://angular.io/>
- [8] React [último acceso: 18/06/2021] Disponible en: <https://es.reactjs.org/>
- [9] Vue [último acceso: 18/06/2021] Disponible en: <https://vuejs.org/>
- [10] Ember [último acceso: 18/06/2021] Disponible en: <https://emberjs.com/>
- [11] Backbone [último acceso: 18/06/2021] Disponible en: <https://backbonejs.org/>
- [12] Meteor [último acceso: 18/06/2021] Disponible en: <https://www.meteor.com/>
- [13] Mithril [último acceso: 18/06/2021] Disponible en: <https://mithril.js.org/>
- [14] TypeScript [último acceso: 18/06/2021] Disponible en: <https://www.typescriptlang.org/>
- [15] ECMA-262 [último acceso: 18/06/2021] Disponible en: <https://www.ecma-international.org/publications-and-standards/standards/ecma-262/>
- [16] Symfony [último acceso: 18/06/2021] Disponible en: <https://symfony.com/>
- [17] Java EE [último acceso: 18/06/2021] Disponible en: <https://www.oracle.com/es/java/technologies/java-ee-glance.html>
- [18] Django [último acceso: 18/06/2021] Disponible en: <https://www.djangoproject.com/>
- [19] ASP.NET [último acceso: 18/06/2021] Disponible en: <https://dotnet.microsoft.com/apps/aspnet>
- [20] XML [último acceso: 18/06/2021] Disponible en: <https://www.w3.org/XML/>
- [21] VBScript [último acceso: 18/06/2021] Disponible en: <https://es.wikipedia.org/wiki/VBScript>
- [22] C# [último acceso: 18/06/2021] Disponible en: https://es.wikipedia.org/wiki/C_Sharp
- [23] VisualBasic.Net [último acceso: 18/06/2021] Disponible en: <http://visualbasic.net/>
- [24] JScript.Net [último acceso: 18/06/2021] Disponible en: https://es.wikipedia.org/wiki/JScript_.NET
- [25] .NET Framework [último acceso: 18/06/2021] Disponible en: <https://docs.microsoft.com/es-es/dotnet/framework/>
- [26] .NET Core [último acceso: 18/06/2021] Disponible en: <https://docs.microsoft.com/es-es/aspnet/core/?view=aspnetcore-5.0>
- [27] NuGet [último acceso: 18/06/2021] Disponible en: <https://www.nuget.org/>
- [28] Visual Studio [último acceso: 18/06/2021] Disponible en: <https://visualstudio.microsoft.com/es/>
- [29] Angular Material [último acceso: 18/06/2021] Disponible en: <https://material.angular.io/>

- [30] SALT (Criptografía) [último acceso: 18/06/2021] Disponible en: [https://es.wikipedia.org/wiki/Sal_\(criptograf%C3%ADa\)](https://es.wikipedia.org/wiki/Sal_(criptograf%C3%ADa))
- [31] Función HASH [último acceso: 18/06/2021] Disponible en: https://es.wikipedia.org/wiki/Funci%C3%B3n_hash
- [32] ¿Qué es y cómo utilizar localStorage y sessionStorage? [último acceso: 18/06/2021] Disponible en: <https://ed.team/blog/que-es-y-como-utilizar-localstorage-y-sessionstorage>
- [33] Angular @Input [último acceso: 18/06/2021] Disponible en: <https://angular.io/api/core/Input>
- [34] EntityLite [último acceso: 18/06/2021] Disponible en: <https://www.codeproject.com/Articles/753796/i-nercya-EntityLite-A-Lightweight-Database-First-M>
- [35] Sistema de gestión de bases de datos [último acceso: 18/06/2021] Disponible en: https://es.wikipedia.org/wiki/Sistema_de_gesti%C3%B3n_de_bases_de_datos
- [36] MySQL [último acceso: 18/06/2021] Disponible en: <https://www.mysql.com/>
- [37] Oracle [último acceso: 18/06/2021] Disponible en: <https://www.oracle.com/>
- [38] PostgreSQL [último acceso: 18/06/2021] Disponible en: <https://www.postgresql.org/>
- [39] Microsoft SQL Server [último acceso: 18/06/2021] Disponible en: <https://www.microsoft.com/es-es/sql-server/sql-server-2019>
- [40] MongoDB [último acceso: 18/06/2021] Disponible en: <https://www.mongodb.com/es>
- [41] Transact-SQL [último acceso: 18/06/2021] Disponible en: <https://es.wikipedia.org/wiki/Transact-SQL>

Glosario

Aplicación web	Herramienta que los usuarios pueden utilizar accediendo a un servidor web a través de internet mediante un navegador
Software	Todo equipo o soporte lógico de un sistema informático que abarca el conjunto de los componentes que son necesarios para la realización de tareas específicas
TS	TypeScript
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
Framework	Esquema para el desarrollo y/o implementación de una aplicación
SPA	Single Page Application
MVC	Modelo Vista Controlador
PHP	HyperText Preprocessor
Java EE	Java Enterprise Edition
ASP	Active Server Pages
XML	Extensible Markup Language
VBScript	Visual Basic Script Edition
IDE	Integrated Development Environment
API	Application Programming Interface
REST	Representational state transfer
SGBD	Sistema Gestión de Base de Datos
ANSI	American National Standards Institute
SQL	Structured Query Language
Pipes	Filtros o funciones que como objetivo tienen convertir un dato en la vista del componente
EPS	Escuela Politécnica Superior
UAM	Universidad Autónoma de Madrid
OPE	Oficina de Prácticas Externas
HASH	Función de derivación de claves
SALT	Bits aleatorios que se usan como una de las entradas, junto con la contraseña, en una función de derivación de claves
Sidenav	Menú deslizable
Namespace	Contenedor para organizar clases dentro de un entorno.
ORM	Object Relational Mapping

Anexos

A Maquetas

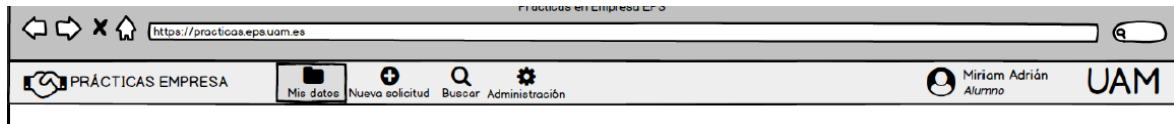


Figura 37. Maqueta *Header*



Figura 38. Maqueta *Nav-menu*

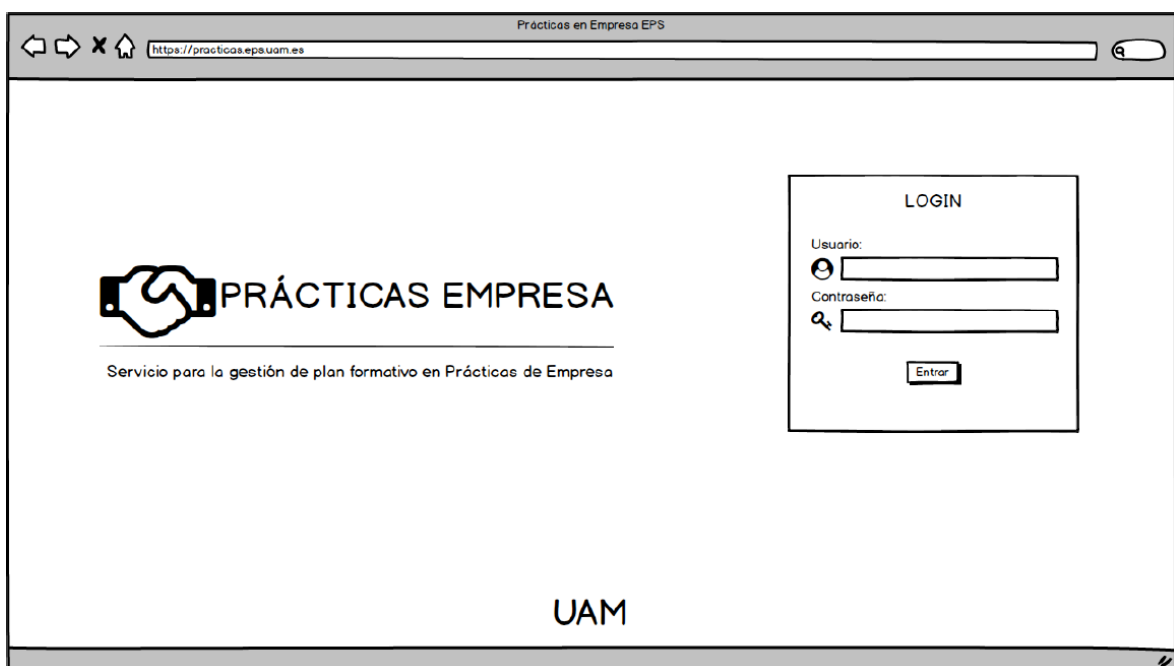


Figura 39. Maqueta *Login*

Prácticas en Empresa EPS

https://practicas.eps.uam.es

PRÁCTICAS EMPRESA

Mis datos Nueva solicitud Buscar Administración

Miriam Adrián Ruiz
ALUMNO

Datos del usuario Mis tareas

Nombre	Miriam Adrián Ruiz	Centro	Escuela Politécnica Superior
Email	miriadrian@estudianteuam.es	Tipo de estudios	Grado
NIA	255530	Plan de estudios	473 - Graduado/a en Ingeniería Informática
DNI	98745632-J	Estudios	2028 - Grado en Ingeniería Informática

Figura 40. Maqueta *Home*

Prácticas en Empresa EPS

https://practicas.eps.uam.es

PRÁCTICAS EMPRESA

Mis datos Nueva solicitud Buscar Administración

Miriam Adrián Ruiz
Alumno

UAM

Nueva solicitud

Guardar Enviar a verificación

Empresa / Proyecto formativo Competencias a obtener Características de la práctica

Empresa

Accenture
Inercya

Tutor Empresarial

Alicia Gómez
Mónica Fernandez

Departamento en el que se realizarán las prácticas

IT

Conocimiento previos por parte del/de la estudiante

Ninguno

Descripción detallada de todas las actividades a realizar por el / la estudiante

Desarrollo de aplicaciones web

Formación a recibir

SQL, Angular, Net Core..

Figura 41. Maqueta *New-request* pestaña *Empresa / Proyecto formativo*

Prácticas en Empresa EPS

https://practicas.eps.uam.es

PRÁCTICAS EMPRESA Mis datos Nueva solicitud Buscar Administración Miriam Adrián Alumno UA

Nueva solicitud

Guardar Enviar a verificación

Empresa / Proyecto formativo Competencias a obtener Características de la práctica

Competencias a obtener

- ☒ Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación y normativa vigente.
- ☐ Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social.
- ☒ Capacidad para comprender la importancia de la negociación, los hábitos de trabajo efectivos, el liderazgo y las habilidades de comunicación en todos los entornos de desarrollo de software.
- ☐ Capacidad para elaborar el pliego de condiciones técnicas de una instalación informática que cumpla los estándares y normativas vigentes.
- ☒ Conocimiento, administración y mantenimiento sistemas, servicios y aplicaciones informáticas.
- ☐ Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.
- ☒ Conocimiento, diseño y utilización de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema.
- ☐ Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
- ☒ Capacidad de conocer, comprender y evaluar la estructura y arquitectura de los computadores, así como los componentes básicos que los conforman.
- ☐ Conocimiento de las características, funcionalidades y estructura de los Sistemas Operativos y diseñar e implementar aplicaciones basadas en sus servicios.
- ☒ Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.
- ☐ Conocimiento y aplicación de las características, funcionalidades y estructura de las bases de datos, que permitan su adecuado uso, y el diseño e implementación de aplicaciones basadas en ellas.
- ☒ Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los sistemas de información, incluidos los basados en web.
- ☐ Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.
- ☒ Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.
- ☐ Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la Ingeniería de software.
- ☒ Capacidad para diseñar y evaluar interfaces persona computador que garanticen la accesibilidad y usabilidad a los sistemas, servicios y aplicaciones informáticas.
- ☐ Conocimiento de la normativa y la regulación de la informática en los ámbitos nacional, europeo e internacional.

Figura 42. Maqueta *New-request* pestaña *Competencias a obtener*

Prácticas en Empresa EPS

https://practicas.eps.uam.es

PRÁCTICAS EMPRESA Mis datos Nueva solicitud Buscar Administración Miriam Adrián Alumno UA

Nueva solicitud

Guardar Enviar a verificación

Empresa / Proyecto formativo Competencias a obtener Características de la práctica

Ctrl+Click to select

Lugar de realización de la práctica

Madrid

Fecha inicio 01/04/2021 **Fecha fin** 31/07/2021

Duración 320

Numero estudiantes que solicita 1

Horas/Día 4

Horario concreto de cada día Mañanas de 10:00 a 14:00

Tipo de práctica Práctica presencial

En caso de semipresencialidad, detallar la planificación

Importe de ayuda al estudio 435€

Persona que firma el anexo Alicia Gómez **Cargo de esa persona** Manager

Dirección Electrónica RGPD alicia.gomez@accenture.es

Figura 43. Maqueta *New-request* pestaña *Características de la práctica*

Prácticas en Empresa EPS

https://practicas.eps.uam.es

PRÁCTICAS EMPRESA

Mis datos Nueva solicitud Buscar Administración

Miriam Adrián Alumno UAM

Detalle de la solicitud

Accenture
Práctica presencial
A - Excelente
01/01/2021 - 30/04/2021

Miriam Adrián Ruiz
Alicia Gómez

Cerrado

Empresa / Proyecto formativo Competencias a obtener Características de la práctica

Empresa
Accenture

Tutor Empresarial
Alicia Gómez

Departamento en el que se realizarán las prácticas
IT

Conocimiento previos por parte del/de la estudiante
Ninguno

Descripción detallada de todas las actividades a realizar por el / la estudiante
Desarrollo de aplicaciones web

Formación a recibir
SQL, Angular, Net Core...

Figura 44. Maqueta *Request* pestaña *Empresa / Proyecto formativo*

C	<p>Accenture</p> <p>Práctica presencial</p> <p>A - Excelente</p> <p>01/01/2021 - 30/04/2021</p>	<p>Miriam Adrián Ruiz</p> <p>Ignacio López</p>	➔
PV	<p>Indra</p> <p>Práctica telemática</p> <p>-</p> <p>01/02/2021 - 31/05/2021</p>	<p>Miguel Rubio Clemente</p> <p>Andrea García</p>	➔
PA	<p>Sopra</p> <p>Práctica presencial</p> <p>C- Deficiente</p> <p>01/05/2021 - 30/09/2021</p>	<p>Alba Martínez Pérez</p> <p>Manuel Sanchez</p>	➔
A	<p>Inercya</p> <p>Práctica presencial y telemática</p> <p>B - Normal</p> <p>01/03/2021 - 31/07/2021</p>	<p>Luis Rodríguez Calvo</p> <p>Jesús Córeres</p>	➔

Figura 45. Maqueta *Request-list*

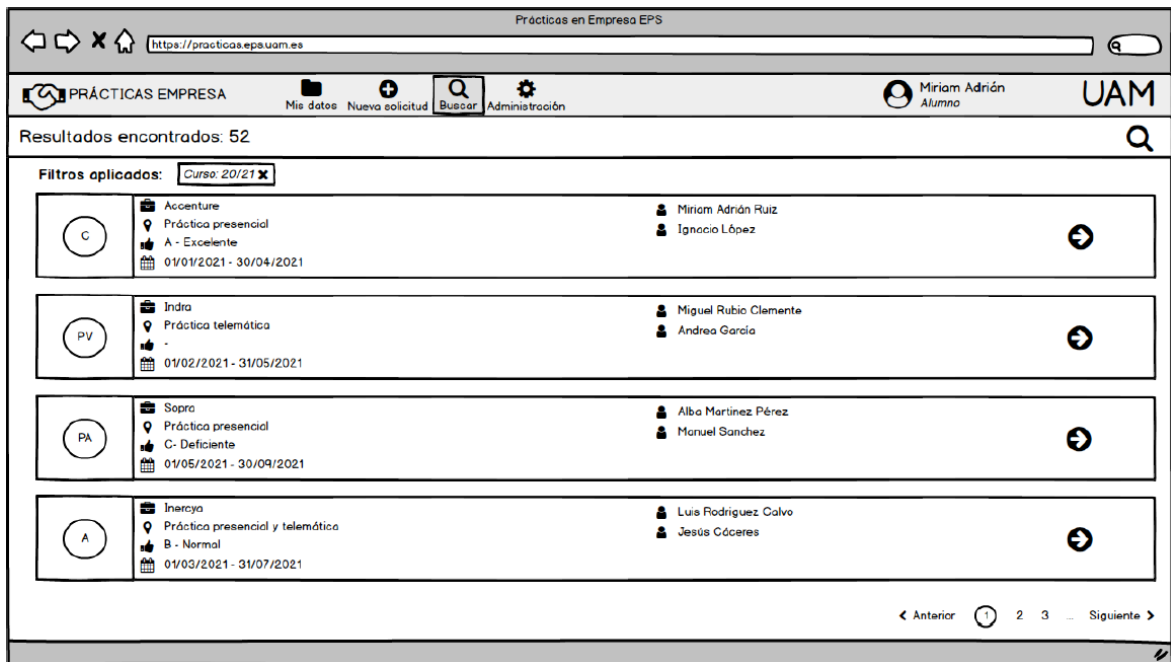


Figura 46. Maqueta *Search*

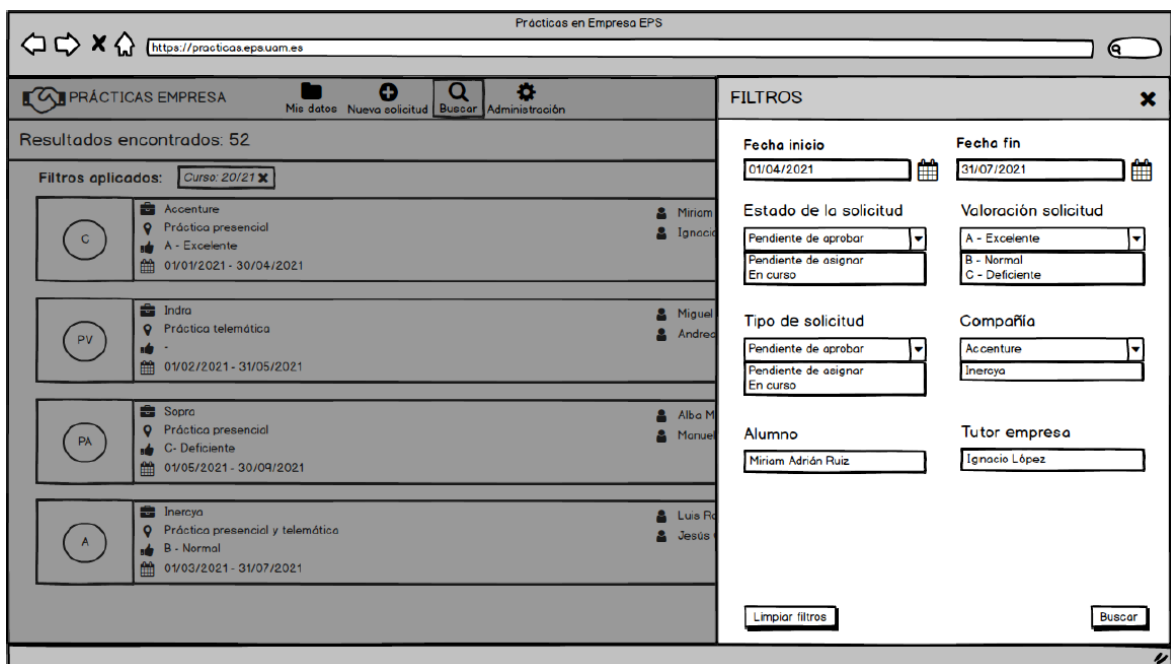


Figura 47. Maqueta *Search* mostrando los filtros incluidos en el *sidebar*

